МИКРОСХЕМА ИНТЕГРАЛЬНАЯ К1892ВМ21Я

РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ

ОГЛАВЛЕНИЕ

1. ВВЕДЕНИЕ 4
2. СТРУКТУРА И ПОДСИСТЕМЫ МИКРОСХЕМЫ 8
3. СИСТЕМНАЯ ОРГАНИЗАЦИЯ МИКРОСХЕМЫ 10
4. ПОДСИСТЕМА ЦЕНТРАЛЬНОГО ПРОЦЕССОРА 64
5. СЛУЖЕБНАЯ ПОДСИСТЕМА 82
6. ПОДСИСТЕМА SDR 118
7. ПОДСИСТЕМА MEDIA 167
8. ПОДСИСТЕМА DDR 203
9. ПОДСИСТЕМА HSPERIPH 224
10. ПОДСИСТЕМА LSPERIPH0 293
11. ПОДСИСТЕМА LSPERIPH1 299
12. ЦЕНТРАЛЬНЫЙ ПРОЦЕССОР (CPU) 309
13. КОНТРОЛЛЕР ПРЕРЫВАНИЙ ЦЕНТРАЛЬНОГО ПРОЦЕССОРА (GIC) 310
14. ГРАФИЧЕСКИЙ ПРОЦЕССОР (GPU) 311
15. БЛОК КОДИРОВАНИЯ И ДЕКОДИРОВАНИЯ ВИДЕО (VPU) 317
16. ПРОЦЕССОР ПРИЕМА И ПРЕДОБРАБОТКИ ИЗОБРАЖЕНИЯ (ISP) 320
17. КОНТРОЛЛЕР ДИСПЛЕЯ (DISPLAY) 347
18. КЛАСТЕР СЛУЖЕБНОГО ПРОЦЕССОРА (RISC) 404
19. СЛУЖЕБНЫЙ КОНТРОЛЛЕР ПРЕРЫВАНИЙ (QLIC) 466
20. НЕЙРОСЕТЕВОЙ АКСЕЛЕРАТОР С ОПТИМИЗАЦИЕЙ ДЛЯ СИГНАТУРНОГО АНАЛИЗА И ОБРАБОТКИ СИГНАЛОВ (DSP) 473
21. БЛОК ЦИФРОВОЙ ОБРАБОТКИ РАДИО (DFE) 498
22. УСКОРИТЕЛИ ЦИФРОВОЙ ОБРАБОТКИ СИГНАЛОВ 688
23. СОПРОЦЕССОР-УСКОРИТЕЛЬ GNSS 764
24. КОНТРОЛЛЕР ДИНАМИЧЕСКОЙ ПАМЯТИ (DDRMC) 786
25. КОНТРОЛЛЕРЫ PCI-EXPRESS (PCIE) И JESD204B 934
26. КОНТРОЛЛЕР USB (USB) 1032
27. КОНТРОЛЛЕР ETHERNET (EMAC) 1184
28. КОНТРОЛЛЕР SD/EMMC (SDMMC) 1283
29. КОНТРОЛЛЕР NAND FLASH (NAND) 1335
30. КОНТРОЛЛЕР QUAD SPI (QSPI) 1389
31. КОНТРОЛЛЕР I2S (I2S) 1431
32. КОНТРОЛЛЕР I2C (I2C) 1453

33. КОНТРОЛЛЕР SSI (SSI) 1524
34. КОНТРОЛЛЕР UART (UART) 1546
35. КОНТРОЛЛЕР ШИМ (PWM) 1591
36. КОНТРОЛЛЕР УНИВЕРСАЛЬНЫХ ВЫВОДОВ (GPIO) 1624
37. КОНТРОЛЛЕР МНОГОФУНКЦИОНАЛЬНОГО БУФЕРИЗИРОВАННОГО ПОСЛЕДОВАТЕЛЬНОГО ПОРТА (MFBSP) С ПОДДЕРЖКОЙ CAN 1634
38. КОНТРОЛЛЕР WATCHDOG TIMER (WDT) 1759
39. БЛОК УНИВЕРСАЛЬНЫХ ТАЙМЕРОВ (TIMERS) 1763
40. ТАЙМЕР РЕАЛЬНОГО ВРЕМЕНИ (RTC) 1769
41. БЛОК ОБМЕНА СООБЩЕНИЯМИ (MAILBOX) 1781
42. БЛОК АТОМАРНЫХ ОПЕРАЦИЙ (SPINLOCK) 1786
43. УНИВЕРСАЛЬНЫЙ БЛОК ФОРМИРОВАНИЯ ТАКТОВОЙ ЧАСТОТЫ (UCG) 1788
44. БЛОК МНОГОПОРТОВОЙ ОБЩЕЙ ПАМЯТИ, ПЕРЕМЕЖИТЕЛЬ (MPSM) 1800
45. УНИВЕРСАЛЬНЫЙ БЛОК ФОРМИРОВАНИЯ СБРОСА (URG) 1813
46. МНОГОКАНАЛЬНОЕ DMA ПАМЯТЬ-ПАМЯТЬ (QDMA) 1817
47. УНИВЕРСАЛЬНОЕ DMA ПЕРИФЕРИЙНЫХ УСТРОЙСТВ (PDMA) 1825
48. БЛОК НАКРИСТАЛЬНЫХ ДАТЧИКОВ (SENSORS) 1881
49. СИСТЕМА ОТЛАДКИ МИКРОСХЕМЫ (DEBUG) 1882
50. КОНТРОЛЛЕР ПРЯМОГО ДОСТУПА В ПАМЯТЬ ПОДСИСТЕМЫ SDR (SDRDMA) 1889
51. ОПИСАНИЕ ВНЕШНИХ ВЫВОДОВ 1903
52. СПРАВОЧНАЯ ИНФОРМАЦИЯ 1961
53. ИСТОРИЯ ИЗМЕНЕНИЙ 1980

1. ВВЕДЕНИЕ

1.1 Состав микросхемы К1892ВМ21Я

Микросхема К1892ВМ21Я («Гиперком У») представляет собой мультиплатформенную систему на кристалле с оптимизированной архитектурой для мультимедийных, навигационных, связных приложений и приложений в робототехнике. Состав микросхемы К1892ВМ21Я представлен на Рисунок 1.1.

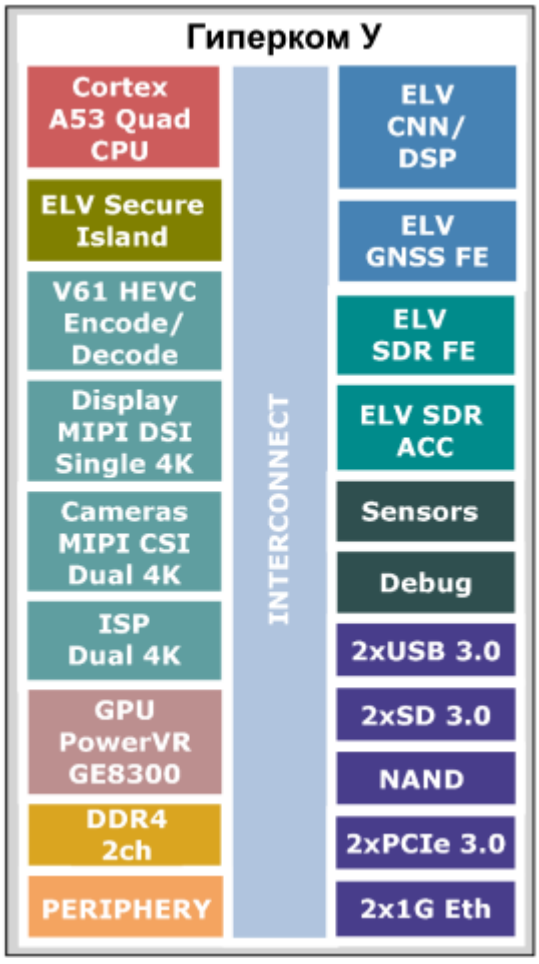


Рисунок 1.1. Состав микросхемы К1892ВМ21Я

Высокопроизводительная СнК К1892ВМ21Я включает 4-х ядерный кластер ARM Cortex-A53, 2-х ядерный нейросетевой акселератор с оптимизацией для сигнатурного анализа и обработки сигналов на базе отечественного DSP ядра ELcore-50 с аппаратной поддержкой сверточных алгоритмов на базе нейросетей, графический процессор PowerVR, аппаратный кодек HEVC/H.264, навигационное ядро с поддержкой ГЛОНАСС/GPS/BeiDou /GALILEO и встроенные высокоскоростные порты ввода/вывода.

Перечень блоков микросхемы, их краткие характеристики и входящие в их состав IP подблоки, представлен в Таблица 1.1.

Таблица 1.1. Блоки в составе СнК микросхемы микропроцессора К1892ВМ21Я

Имя	Описание	IP Подблоки
CPU	Процессор общего назначения 4-х ядерный кластер ARM Cortex-A53 MPCore • тактовая частота ядра - не менее 1.12 ГГц[1]; • встроенный L1 кэш - 32 Кбайта; • L2 кэш - 1 Мбайт; • SIMD/FPU сопроцессоры NEON для каждого из ядер.	
GPU	Графическое ядро GPU • тактовая частота - не менее 500 МГц[1]; • поддержка OpenGL ES 3.2, OpenCL 1.2 EP, OpenVX 1.1; • OpenCL API; • поддержка Vulkan 1.0;	
DSP0 DSP1	Нейросетевой акселератор с оптимизацией для сигнатурного анализа и обработки сигналов на базе кластера из двух DSP ядер ELcore50: • тактовая частота ядра- не менее 340 МГц[1]; • поддержка 8/16/32/64 - разрядных данных с фиксированной точкой • поддержка 16/32/64 - разрядных данных с плавающей точкой; • C - компилятор; • встроенный L1 кэш - 16 Кбайт; • встроенный L2 кэш/быстрая память - 512 Кбайт; • набор инструкций для работы с CNN;	
RISC0 RISC1	Управляющее и служебное процессорное ядро • 32 битная MIPS архитектура; • встроенный L1 кэш – 32 Кбайта инструкций, 32 Кбайта данные; • тактовая частота - не менее 560 МГц при напряжении источника питания ядра потребления (UCC2=0,9 В ±5%)[1]; • встроенная память быстрого доступа объемом 32 Кбайта; • JTAG отладчик OnCD;	CRAM ROM
VPU	HEVC/H.264 ядро кодера/декодера ARM Mali V61: • формат данных: 10/8 бит 4.2.2 и 4.2.0; • поддержка 2 потоков 4K@60 кодирования/декодирования или 1 потока 4K@60 кодирования плюс 1 потока 4K@60 декодирования; • поддержка JPEG/MPEG;	
ISP	Ядро предобработки изображения и видеоввода: • поддержка 2-х потоков 4K@30 или 1 потока 4K@60; • кадрирование, децимация, позиционирование; • коррекция пикселей; • поддержка HDR; ○ line-by-line и pixel-by-pixel HDR для двух экспозиций; ○ frame based HDR с программным выполнением HDR merge; • масштабирование; • функции статистики.	MIPI DPHYRx0 MIPI DPHYRx1
DISPLAY	Порт видео вывода	MIPI DSI CTR

Имя	Описание	IP Подблоки
	• интерфейс MIPI DSI; • параллельный интерфейс RGB; • поддержка 1 потока 4K@30 (при использовании DSI интерфейса).	MIPI DPHYTx
GNSS	Навигационное ядро GNSS: • поддержка 4-х стандартов: ГЛОНАСС/GPS/BeiDou/GALILEO; • внешний RF модуль; • поставляемое ПО;	
DDRMC0 DDRMC1	Контроллеры DDR памяти • разрядность 32 бита на контроллер; • скорость не менее 1600 Мбит/с на каждую линию; • поддержка форматов DDR3/ LPDDR3/ DDR4/ LPDDR4; • inline ECC;	DDR PHY0 DDR PHY1
USB0 USB1	Контроллеры USB 3.0 DRD;	USB PHY0 USB PHY1
PCIe0 PCIe1	Контроллеры PCIe: • конфигурация линий 2 x 4; • поддержка PCI Express 3.0; • скорость на линию 8Гбит/с; • поддержка и RootComplex и EndPoint.	16G PHY0[2)] 16G PHY1[3)]
QDMA0 QDMA1	Четырехканальное DMA память-память • двумерная адресация; • самоинициализация;	
SDRDMA0 SDRDMA1	Восьмиканальное DMA память-память • двумерная адресация; • самоинициализация;	
PDMA0 PDMA1 PDMA2	DMA для периферийных устройств	
UART0 UART1 UART2 UART3	Контроллеры UART	
I2S	Контроллер I2S, 2-х канальный аудиовход, 4-х канальный аудиовыход	
I2C0 I2C1 I2C2 I2C3	Контроллеры I2C	
SSI0 SSI1	Контроллеры SSI	
NAND	Контролер NAND Flash памяти: • 16 бит; • поддержка ONFI 3.2	
SDMMC0 SDMMC1	Контроллеры SD/eMMC 4.5	
QSPI0 QSPI1	Контроллеры QuadSPI с функцией непосредственного исполнения XiP	
PWM	ШИМ контроллер	
MAILBOX0	Блоки «почтовых ящиков» для межпроцессорного взаимодействия	

Имя	Описание	IP Подблоки
MAILBOX1		
SPINLOCK	Блок поддержки атомарных операций	
WDT0 WDT1	Сторожевые таймеры	
TIMERS0 TIMERS1	Блоки универсальных таймеров	
GPIO0 GPIO1	Блоки универсальных выводов по 32 разряда	
MIPI DSI	Контроллер интерфейса видеовывода MIPI DSI 1.2	
MIPI CSI0 MIPI CSI1	Контроллер интерфейса видеоввода MIPI CSI2 1.2	
MFBSP0 MFBSP1	Универсальные контроллеры с функциями LPORT, SSI, I2S, GPIO, CAN;	
ACC	Кластер аппаратных акселераторов обработки сигналов • декодер Витерби • сверточный кодер • турбо кодер • турбо декодер • блок быстрого преобразования Фурье	
DFE0 DFE1	Блоки цифрового фронтэнда для реализации SDR	JESD204B0 JESD204B1
JESD204B0 JESD204B1	Контроллеры интерфейса JESD204B	
SPRAM0	Блоки накристальной памяти общего назначения в составе подсистемы HSPERIPH, объемом 64 Кбайта	
MPSM	Блок накристальной памяти общего назначения в составе подсистемы SDR, объемом 4 Мбайта	
SDRDMA	Специализированный блок DMA память-память в составе подсистемы SDR	
GIC	Контроллер прерывания для CPU	
EMAC0 EMAC1	Контроллеры 1G RGMII Ethernet	
SENSORS	Блок накристальных датчиков температуры и напряжения	
OTP	Блок энергонезависимой однократно программируемой памяти 4 Кбит	
TRNG	Блок генератора случайных чисел	
DEBUG	Система отладки и мониторинга микросхемы через USB или JTAG	

[1)] *Значения тактовой частоты в данной таблице приведены для нормальных условий эксплуатации, в остальной части документа приведены значения для работы для наихудших условий.*

[2)] *PHY является разделяемым для функций PCIe0 и JESD204B0, и может быть использован только в одном из этих режимов*

[3)] *PHY является разделяемым для функций PCIe1 и JESD204B1, и может быть использован только в одном из этих режимов*

2. СТРУКТУРА И ПОДСИСТЕМЫ МИКРОСХЕМЫ

Микросхема структурно разбита на несколько подсистем, показанных на Рисунок 2.1.

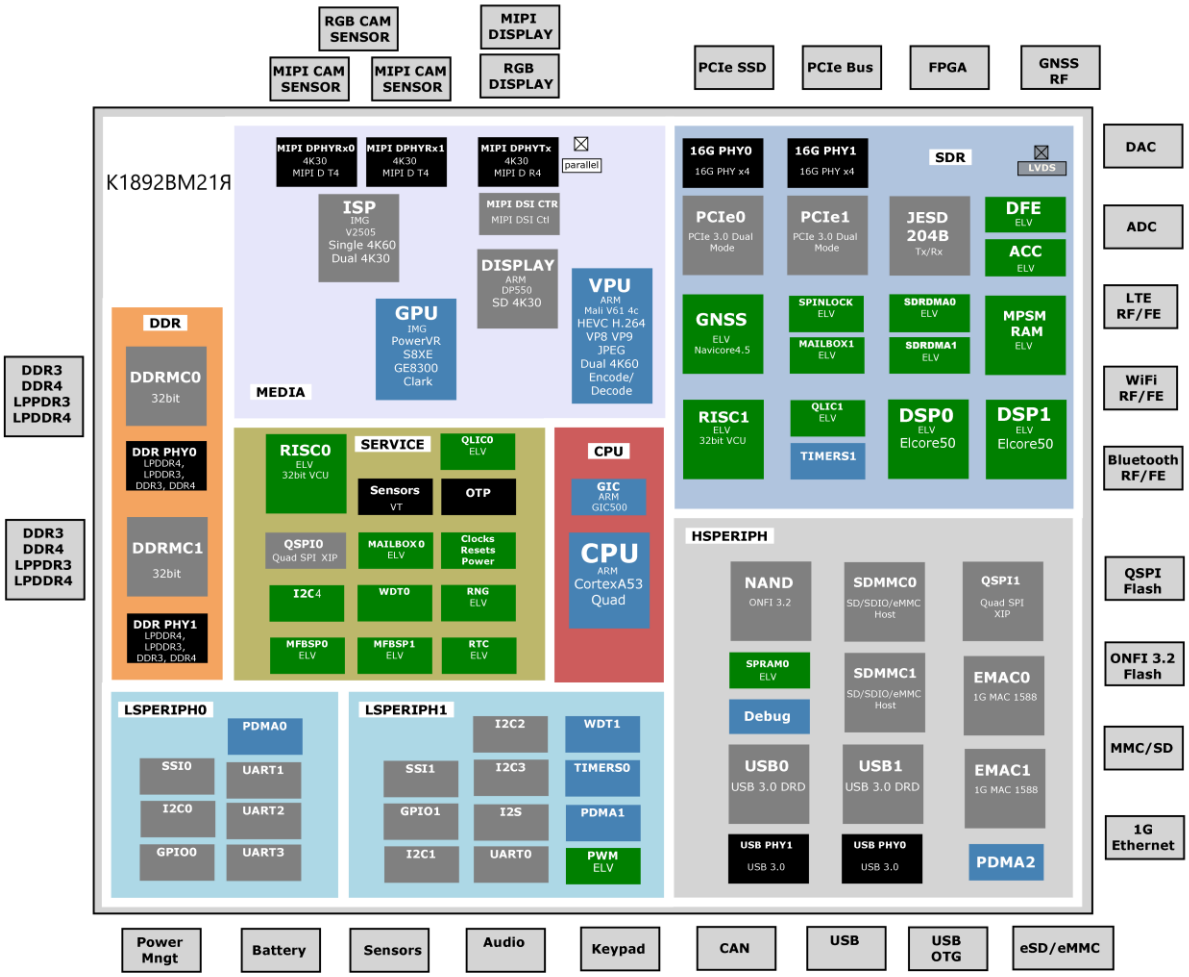


Рисунок 2.1. Структурная схема микросхемы

Подсистемы объединяют в себя функционально связанные блоки, и включают логику и регистры их общего управления, собственные локальные системы коммутации, формирования сброса, формирования частот, в частности один или несколько блоков PLL, а также контактные площадки блоков и логику их управления.

Список подсистем, их назначение и составляющие их блоки указаны в Таблица 2.1.

Таблица 2.1. Список подсистем и блоков микросхемы

Подсистема	Назначение	Блоки
sdr_subs	Подсистема SDR	PCIe0, PCIe1, JESD204B0, JESD204B1, DFE0, DFE1, GNSS, ACC, MPSM, DSP0, DSP1, QLIC1, RISC1, SPINLOCK1, MAILBOX1, TIMERS1, SDRDMA0, SDRDMA1

service_subs	Служебная подсистема	RISC0, SENSORS, QLIC0, OTP, QSPI0, WDT0, TRNG, MFBSP0, MFBSP1, RTC, MAILBOX0, I2C4
ddr_subs	Подсистема портов динамической памяти	DDR0, DDR1
cpu_subs	Подсистема центрального процессора	CPU, GIC
media_subs	Подсистема мультимедиа	ISP, DISPLAY, GPU, VPU, MIPI CSI0, MIPI CSI1, MIPI DSI
hsperiph_subs	Подсистема высокоскоростной периферии	NAND, SDMMC0, SDMMC1, QSPI1, SPRAM0, DEBUG, EMAC0, EMAC1, USB0, USB1, PDMA2
lsperiph0_subs	Подсистема низкоскоростной периферии 0	PDMA0, SSI0, I2C0, GPIO0, UART1, UART2, UART3
lsperiph1_subs	Подсистема низкоскоростной периферии 1	WDT1, TIMERS0, PDMA1, UART0, I2C1, I2C2, I2C3, I2S, GPIO1, SSI1, PWM

Базовой подсистемой микросхемы, т.е. подсистемой ответственной за формирование опорных частот, сброса, управления питанием для других подсистем и содержащей логику и регистры системных настроек микросхемы, а также блоки, работающие при старте и начальной загрузке микросхемы является служебная подсистема.

Система коммутации верхнего уровня, в виде набора межсоединений и коммутаторов, логики и регистров их настройки, логики формирования частот и сброса также выделена в отдельную подсистему.

3. СИСТЕМНАЯ ОРГАНИЗАЦИЯ МИКРОСХЕМЫ

3.1 Карта памяти микросхемы

Карта физической памяти микросхемы приведена в Таблица 3.1. Здесь и далее, если это не оговорено специально, коды адреса и данных указаны в шестнадцатеричной системе счисления.

Размер физического адреса в микросхеме равен 40 битам.

Таблица 3.1. Карта физической памяти

Базовый адрес	Конечный адрес	Описание	Размер		
		Максимальный размер адреса:	*32 бит*	*36 бит*	*40 бит*
C0_0000_0000	FF_FFFF_FFFF	PCIe1 High			256 Гбайт
80_0000_0000	BF_FFFF_FFFF	PCIe0 High			256 Гбайт
10_0000_0000	7F_FFFF_FFFF	Зарезервировано			448 Гбайт
08_0000_0000	0F_FFFF_FFFF	DDR High		32 Гбайта	
07_8000_0000	07_FFFF_FFFF	DDR Low Mirror (нет прямого доступа)		2 Гбайта	
05_C000_0000	07_7FFF_FFFF	PCIe1 Mid		7 Гбайт	
04_0000_0000	05_BFFF_FFFF	PCIe0 Mid		7 Гбайт	
01_0000_0000	03_FFFF_FFFF	Зарезервировано		12 Гбайт	
00_8000_0000	00_FFFF_FFFF	DDR Low	2 Гбайта		
00_7000_0000	00_7FFF_FFFF	PCIe1 Low	256 Мбайт		
00_6000_0000	00_6FFF_FFFF	PCIe0 Low	256 Мбайт		
00_5000_0000	00_5FFF_FFFF	QuadSPI (QSPI1)	256 Мбайт		
00_4000_0000	00_4FFF_FFFF	QuadSPI (QSPI0)[1)]	256 Мбайт		
00_0000_0000	00_3FFF_FFFF	Внутренняя память	1 Гбайт		

Прямой доступ к внешней памяти может осуществляться через порты DDRMC0, DDRMC1, QSPI0, QSPI1.

[1)]Доступ к порту QSPI0 в режиме XiP возможен только для мастеров в составе служебной подсистемы.

Порты памяти DDRMC0, DDRMC1 предназначены для доступа к памяти типа DDR3, DDR3L, DDR4, LPDDR3, LPDDR4. Адресуемое пространство DDR разбито на два диапазона – старший DDR High и младший DDR Low. Для этих диапазонов доступно два режима адресации.

В прямом режиме диапазоны разбиваются на два контроллера поровну, как это показано в Таблица 3.2.

Таблица 3.2. Прямая адресация данных в DDR

Диапазон адресов		Контроллер DDR
DDR High		
0C_0000_0000	0F_FFFF_FFFF	DDRMC1

08_0000_0000	0B_FFFF_FFFF	DDRMC0
DDR Low		
00_C000_0000	00_FFFF_FFFF	DDRMC1
00_8000_0000	00_BFFF_FFFF	DDRMC0

В режиме чередования данные в DDR чередуются между контроллерами постранично с шагом в 4 Кбайта, как это показано в Таблица 3.3.

Таблица 3.3. Чередование данных в DDR

Пример	Паттерн чередования	Контроллер DDR
DDR High		
08_0000_1000	Addr[39:38] = 0x0 Addr[37] = 0x1 Addr[12] = 0x1	DDRMC1
08_0000_0000	Addr[39:38] = 0x0 Addr[37] = 0x1 Addr[12] = 0x0	DDRMC0
DDR Low		
00_8000_1000	Addr[39:32] = 0x00 Addr[31] = 0x1 Addr[12] = 0x1	DDRMC1
00_8000_0000	Addr[39:32] = 0x00 Addr[31] = 0x1 Addr[12] = 0x0	DDRMC0

Порты память QSPI0, QSPI1 предназначены доступа к памяти типа Serial NOR Flash. Прямая адресация памяти через эти порты возможна только по чтению.

Адресация данных в областях внешней памяти допускается как с точностью до слова, так и с точностью до байта.

Адресация данных в областях внутренней памяти допускается только с точностью до слова. Байтовые обращения в области внутренней памяти не допускаются и могут привести к непредсказуемым последствиям.

Программный доступ к резервным областям запрещен, это может привести к непредсказуемым последствиям. Карта внутренней памяти приведена в таблице Таблица 3.4.

Таблица 3.4. Карта внутренней памяти

Стартовый адрес	Размер области	Описание	Подсистема, порт	Уровень доступа
0x0000000000	64 Кбайта	SPRAM0	hsperiph, sys	non-secure, securable
0x0000010000	None	Резерв	-	-
0x0000800000	1 Мбайт	Регистры интерфейса GIC CPU	только для cpu	-
0x0000900000	3 Мбайта	Резерв	-	-
0x0000c00000	4 Мбайта	Отладочная область и отладочные регистры CPU	cpu, sys	trusted, securable

0x0001000000	1 Мбайт	Регистры блоков подсистемы cpu_subs	cpu, sys	trusted, securable
0x0001100000	1 Мбайт	Блок GIC500 подсистемы cpu_subs	cpu, sys	non-secure, securable
0x0001200000	1 Мбайт	Регистры блоков подсистемы media_subs	media, sys	non-secure, securable
0x0001300000	64 Кбайта	Регистры Display Processor.	media, sys	non-secure, securable
0x0001310000	64 Кбайта	Регистры Display Processor	media, sys	secure, non-securable
0x0001320000	64 Кбайта	Область регистров настройки подсистемы media_subs	media, sys	trusted, securable
0x0001330000		Резерв		
0x0001600000	512 Кбайт	Регистры блоков подсистемы lsperiph0_subs	lsperiph0, sys	non-secure, securable
0x0001680000	512 Кбайт	Область регистров настройки подсистемы lsperiph0_subs	lsperiph0, sys	trusted, securable
0x0001700000	768 Кбайт	Регистры блоков подсистемы lsperiph1_subs	lsperiph1, sys	non-secure, securable
0x00017c0000	256 Кбайт	Область регистров настройки подсистемы lsperiph1_subs	lsperiph1, sys	trusted, securable
0x0001800000	64 Кбайта	Регистры системы коммутации и уровня top	top	trusted, securable
0x0001810000		Резерв	-	-
0x0001900000	1 Мбайт	Регистры подсистемы sdr_subs	sdr, sys	sdr, securable
0x0001a00000	1 Мбайт	Регистры настойки PHY PCIe0 sdr_subs	sdr, sys	sdr, securable
0x0001b00000	2 Мбайта	Регистры подсистемы sdr_subs	sdr, sys	sdr, securable
0x0001d00000	1 Мбайт	Регистры настойки PHY PCIe1 sdr_subs	sdr, sys	sdr, securable
0x0001e00000	16 Кбайт	MAILBOX1 подсистемы sdr_subs регион 0	sdr, sys	non-secure
0x0001e04000	16 Кбайт	MAILBOX1 подсистемы sdr_subs регион 1	sdr, sys	secure
0x0001e08000	4 Кбайта	MAILBOX1 Soft Reset	sdr, sys	sdr, securable
0x0001e09000	-	Резерв	sdr, sys	sdr, securable
0x0001f00000	1 Мбайт	Регистры подсистемы sdr_subs	sdr, sys	sdr, securable
0x0002000000	4 Мбайта	Регистры настойки контроллера PCIe0 sdr_subs	sdr, sys	sdr, securable
0x0002400000	4 Мбайта	Регистры настойки контроллера PCIe0 sdr_subs	sdr, sys	sdr, securable
0x0002800000	8 Мбайт	Резерв	-	-
0x0003000000	16 Мбайт	Внутренняя память подсистемы sdr_subs	sdr, dev	sdr, securable
0x0004000000	64 Мбайта	Регистры DDR0 подсистемы ddr_subs	ddr, sys	trusted, securable
0x0008000000	64 Мбайта	Регистры DDR1 подсистемы ddr_subs	ddr, sys	trusted, securable
0x000c000000	64 Мбайта	Область регистров настройки подсистемы ddr_subs	ddr, sys	trusted, securable
0x0010000000	4 Мбайта	Регистры блоков подстистемы hsperiph_subs	hsperiph, sys	non-secure, securable
0x0010400000	4 Мбайта	Область регистров настройки подсистемы hsperiph_subs	hsperiph, sys	trusted, securable
0x0010800000	-	Резерв	-	-
0x001efd0000	16 Кбайт	MAILBOX0 подсистемы	service, sys	non-secure

		service_subs регион 0		
0x001efd4000	8 Кбайт	MAILBOX0 подсистемы service_subs регион 1	service, sys	secure
0x001efd6000	8 Кбайт	MAILBOX0 подсистемы service_subs регион 2	service, sys	sdr
0x001efd8000	4 Кбайта	MAILBOX0 Soft Reset	service, sys	trusted, securable
0x001efd9000	-	Резерв	service, sys	trusted, securable
0x001f000000	16 Мбайт	Регистры и память подсистемы service_subs	service, sys	trusted, securable
0x0020000000	-	Резерв	service, sys	trusted, securable
0x0040000000	256 Мбайт	QSPI0 подсистемы service_subs[1)]	service, sys	trusted
0x0050000000	256 Мбайт	QSPI1 подсистемы hsperiph_subs	hsperiph, sys	non-secure, securable
0x0060000000	256 Мбайт	PCIe0 Low	sdr, dev	non-secure, securable
0x0070000000	256 Мбайт	PCIe1 Low	sdr, dev	non-secure, securable
0x0080000000	512 Мбайт	DDR Low Trusted 0	ddr	trusted, securable
0x00a0000000	512 Мбайт	DDR Low SDR 0	ddr	sdr, securable
0x00c0000000	512 Мбайт	DDR Low Trusted 1	ddr	trusted, securable
0x00e0000000	512 Мбайт	DDR Low SDR 1	ddr	sdr, securable
0x0100000000	12 Гбайт	Резерв	-	-
0x0400000000	7 Гбайт	PCIe0 Mid	sdr, dev	trusted, securable
0x05C0000000	7 Гбайт	PCIe1 Mid	sdr, dev	trusted, securable
0x0780000000	512 Мбайт	DDR Low Mirror Trusted 0	ddr	trusted, securable
0x07a0000000	512 Мбайт	DDR Low Mirror SDR 0	ddr	sdr, securable
0x07c0000000	512 Мбайт	DDR Low Mirror Trusted 1	ddr	trusted, securable
0x07e0000000	512 Мбайт	DDR Low Mirror SDR 1	ddr	sdr, securable
0x0800000000	32 Гбайта	DDR High	ddr	non-secure, securable
0x1000000000	448 Гбайт	Резерв	-	-
0x8000000000	256 Гбайт	PCIe0 High	sdr, dev	non-secure
0xC000000000	256 Гбайт	PCIe1 High	sdr, dev	non-secure

[1)]Доступ к порту QSPI0 в режиме XiP возможен только для мастеров в составе служебной подсистемы.

В Таблица 3.4. Карта внутренней памятиданы лишь базовые адреса устройств; полные перечни программно-доступных регистров для каждого устройства приведены в главах документа описывающих эти устройства.

3.1.1 Доступ к памяти DDR High для 32 битных устройств

В микросхеме присутствуют ряд устройств мастеров, поддерживающих только 32-битную адресацию и максимальное физическое адресное пространство в 4 Гбайта, и при этом не имеющих собственных устройств или механизмов трансляции физического адреса, таких как VMMU для RISC0 и RISC1. В числе этих устройств - периферийные DMA – PDMA0, PDMA1, PDMA2; DMA встроенные в контроллеры EMAC0 и EMAC1; и графический процессор GPU, входящие в состав подсистем lsperiph0, lsperiph1, hsperiph и media.

Для этих устройств можно настроить смещение физического диапазона в 4 Гбайта (0x00_0000_0000 – 0x00_FFFF_FFFF), адресуемого устройством, в области памяти старше адреса 0x00_FFFF_FFFF. См. системные регистры GPU_BAR, HSPERIPH_BAR, LSPERIPH0_BAR, LSPERIPH1_BAR ddr_subs. Для того чтобы при активном смещении

сохранить доступ этих устройств к диапазону памяти DDR Low (0x00_8000_0000 – 0x00_FFFF_FFFF), в карте памяти предусмотрен зеркальный диапазон DDR Low Mirror (0x07_8000_0000 – 0x07_FFFF_FFFF), при обращении к которому, запросы перенаправляются в диапазон DDR Low коммутатором микросхемы.

3.2 Система обеспечения безопасности в микросхеме

3.2.1 Следование концепциям обеспечения безопасности

Система обеспечения безопасности микросхемы следует в посильной мере концепции Trusted Execution Environment (TEE) компании GlobalPlatform, описанной в документе TEE System Architecture Version 1.1; и концепции Trusted Base System Architecture (TBSA) компании ARM, описанной в документе Trusted Base System Architecture, Client, 3rd Edition.

Также микросхема аппаратно совместима с технологией ARM TrustZone.

Микросхема разделена на три контура безопасности – доверенный, связной и общий, показанные на Рисунок 3.1.

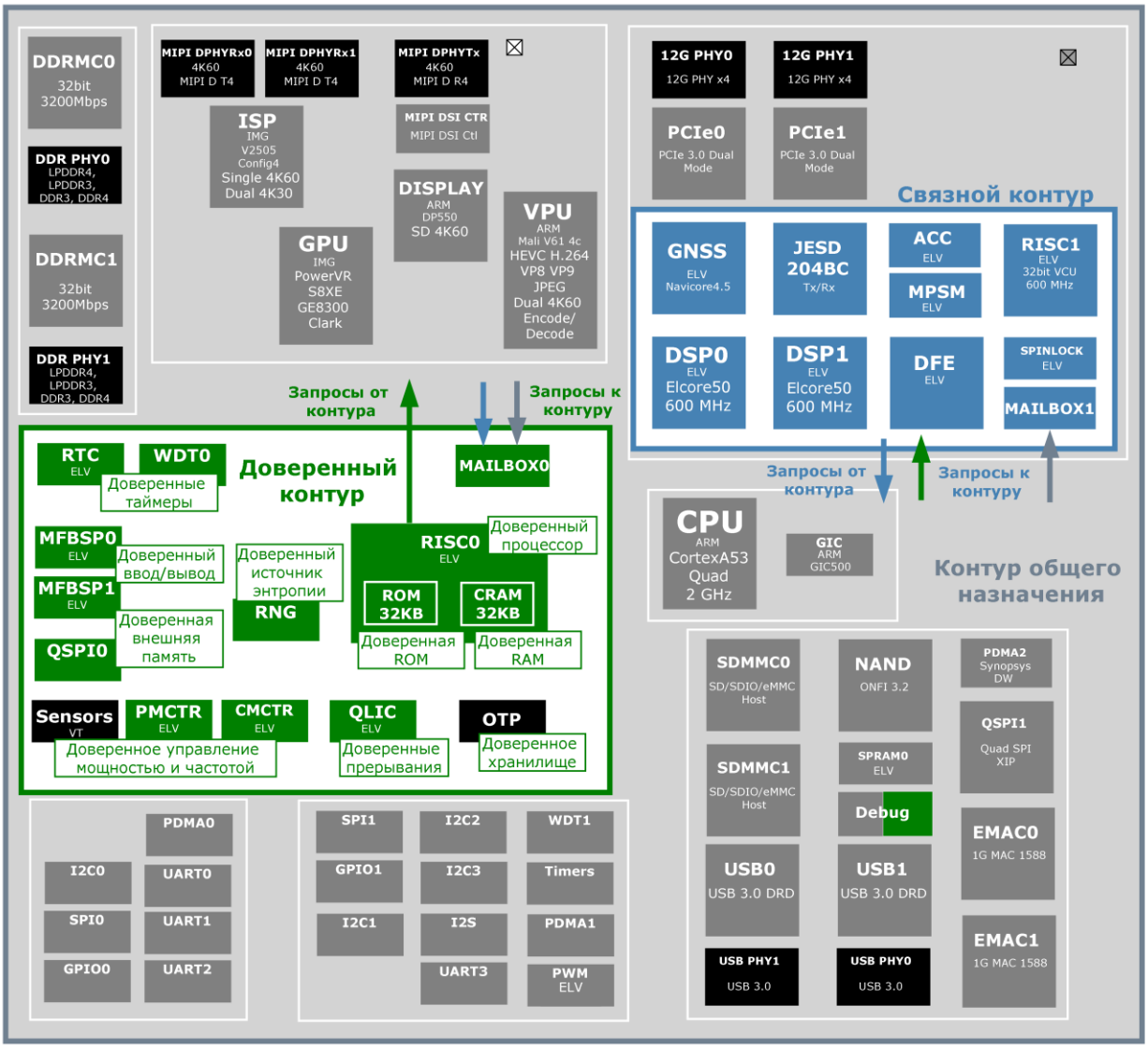


Рисунок 3.1. Контуры безопасности в микросхеме

Доверенный контур является основной аппаратной базой для доверенной среды исполнения (Trusted Execution Environment – TEE). RISC0 – процессор, на котором исполняется доверенная ОС, предоставляющая сервисы для ПО общего и связного назначения, исполняющемся в общем и связном контуре. Базой для исполнения сервисов являются следующие аппаратные компоненты контура:

- RISC0 – доверенное процессорное ядро;
- QLIC0 – контроллер доверенных прерываний;
- OTP – доверенное хранилище ключевой информации;
- RTC – доверенный таймер реального времени;
- WDT0 – доверенный watchdog таймер;
- RNG – доверенный генератор случайных чисел;
- Sensors – накристальные сенсоры температуры и напряжения;
- PMCTR – доверенный контроллер управления мощностью;
- CMCTR – доверенный корневой контроллер управления системой синхронизации;
- ROM – доверенная накристальная ROM память;
- CRAM – доверенная накристальная CRAM память;
- MFBSP0 – доверенный контроллер периферии или SPI Flash памяти;
- MFBSP1 – доверенный контроллер периферии или SPI Flash памяти;
- QSPI0 – доверенный контроллер SPI Flash памяти с функцией непосредственного исполнения;

Контур общего назначения является аппаратной базой для пользовательской среды исполнения (Rich Execution Environment – REE). CPU – процессор, на котором исполняется ОС общего назначения (Linux, Android).

Связной контур является аппаратной базой для среды исполнения связного ПО.

Для разграничения контуров безопасности реализованы аппаратные фильтры доступа, работающие следующим образом:

- Любые запросы в доверенный контур от общего или связного контура разрешены только через доверенный почтовый ящик MAILBOX0.
- Любые запросы в связной контур от общего контура разрешены только через почтовый ящик MAILBOX1.
- Запросы от доверенного контура имеют прямой неограниченный доступ в связной и общий контур.

- Запросы от связного контура имеют прямой неограниченный доступ в общий контур.

Примечание: описание полного программно-аппаратного комплекса для обеспечения безопасности и концепции безопасности, реализованной на его основе, выходит за рамки данного документа. В состав комплекса как правило входят: доверенная ОС (базовый фреймворк, агент коммуникации с REE, API для приложений, драйверы доверенных устройств), доверенные приложения, и доверенные аппаратные компоненты. Здесь описаны лишь последние – аппаратные компоненты, на основе которых такой комплекс может быть реализован.

3.2.2 Безопасность в системе коммутации микросхемы

Существует несколько уровней доступа и соответственно уровней запросов в системе коммутации микросхемы.

В порядке убывания уровня доступа:

- Доверенный уровень (trusted) – запросы от доверенного контура и внутри него от его компонентов.
- Связной уровень (sdr) – запросы от связного контура и внутри него от его компонентов.
- Безопасный уровень (secure) – запросы от компонентов общего контура, в рамках архитектуры ARM TrustZone помеченных как безопасные.
- Общий уровень (non-secure) – запросы от компонентов общего контура, в рамках архитектуры ARM TrustZone помеченные как небезопасные.

Области памяти имеют такие же уровни по доступу, отмечающие какого уровня запросы допустимы к этой области. Более высокий уровень доступа для запроса означает, что для него допустимы обращения к областям памяти с уровнями равными, либо ниже его. Т.е. для запросов trusted допустимы обращения к областям trusted, sdr, secure и non-secure, для запросов sdr к областям sdr, secure, non-secure и т.д.

Для областей памяти существует также особый признак, касающийся доступа – securable, это означает что области памяти, изначально по сбросу, имеющей другой уровень доступа можно присвоить уровень доступа secure. Это работает как в случае повышения уровня, когда области памяти с уровнем доступа non-secure можно сделать secure, так и в случае понижения уровня – некоторые области доверенного (trusted) и связного (sdr) контура можно сделать доступными для secure обращений. Регистры управления для этого (см. регистры NONESECURETOSECURE, DDR_LOW_RANGE_SECURE_CTR,

SDRTOSECURE и TRUSTEDTOSECURE Таблица 3.21 в Разделах 3.9.2.3, 3.9.2.4, 3.9.2.5 и 3.9.2.6 соответственно) доступны только для доверенных обращений.

В регистре FUSE0 подсистемы service_subs существует fuse-бит TRUSTEDTOSECURE_DISABLE, запрещающий изменение прав доступа с trusted на secure. После прожига этого бита признак securable становится не применим к областям с доверенным доступом (trusted, sdr).

Разграничение областей памяти по уровням доступа и наличие признака securable показано в таблице Таблица 3.4 – карта внутренней памяти микросхемы.

Аппаратно, на уровне системы коммутации микросхемы, разграничение запросов от порта мастера производится следующим образом:

- trusted – AxPROT[1] == 0 и определенный ID;
- sdr – AxPROT[1] == 0 и определенный ID;
- secure – AxPROT[1] == 0;
- non-secure – AxPROT[1] == 1.

Диапазоны ID для контуров trusted и sdr определены в регистрах TRUSTED_ID_x, SDR_ID_x, см Разделы 3.9.2.7 и 3.9.2.8. Полный список ID всех мастеров микросхемы указан в Таблица 3.41.

3.2.3 Взаимодействие с доверенным и связным контуром

Доверенный почтовый ящик – MAILBOX0 позволяет устройствам в общем или связном контуре взаимодействовать с доверенным контуром. Он позволяет организовать как входящие запросы, так и ответы на них. При посылке запроса или сообщения из общего или связного контура в доверенный, производится запись данных о запросе и установка запроса на прерывание в почтовом ящике. Прерывание, через контроллер прерываний QLIC0, транслируется в доверенный процессор RISC0. По готовности ответа, доверенный контур записывает данные о нем в почтовый ящик, вместе с генерацией другого – ответного прерывания, которое транслируется через контроллер прерываний общего контура GIC500 в CPU, либо через контроллер прерываний связного контура QLIC1 в RISC1 или DSP.

Похожим образом организовано взаимодействие общего контура со связным контуром. Через почтовый ящик MAILBOX1, он связан прерываниями, с одной стороны, с процессорами RISC1 и DSP, с другой – с процессором CPU.

MAILBOX0 имеет три набора регистров для генерации прерываний в доверенный контур – с уровнем доступа secure, non-secure и sdr.

MAILBOX1 имеет два набора регистров для генерации прерываний в связной контур – с уровнем доступа secure и non-secure.

3.2.4 Доверенный процессор RISC0

Процессор RISC0 является основным элементом доверенного контура и выполняет следующие задачи:

- Доверенная ОС, службы и приложения – Trusted Execution Environment (TEE);
- Криптоалгоритмы. Довереный контур не содержит аппаратных криптоакселераторов, поэтому все криптографические алгоритмы исполняются программно, на процессоре.

RISC0 поддерживает архитектуру MIPS, содержит кэши первого уровня для инструкций и данных, сопроцессор FPU, MMU с TLB и встроенные памяти – быстрого доступа CRAM и ROM. Кроме того, в кластер процессора входит DMA память-память для доверенных пересылок и дополнительное устройство преобразования адресов VMMU для расширения физического адреса из пространства 2^{32} (MIPS32) в пространство 2^{40}(физическая память микросхемы) и виртуализации.

Прерывания в доверенный процессор поступают через выделенный доверенный контроллер прерываний QLIC0.

3.2.5 Безопасность процессора общего назначения CPU

Процессор CPU общего контура – кластер Cortex A53 MPCore поддерживает архитектуру ARMv8-A, включая Security Extensions. Подробнее в документе – ARM® Cortex®-A53 MPCore Processor Technical Reference Manual.

Процессор CPU может быть инициатором запросов уровней secure и non-secure.

3.2.6 Безопасность в контроллере прерываний общего назначения GIC500

Контроллер прерываний общего контура GIC500 поддерживает архитектуру ARM GICv3 со следующими свойствами в отношении безопасности:

- три группы безопасности для прерываний – две для безопасных прерываний (secure) и одна для небезопасных (non-secure);
- любое прерываний может быть назначено одной из групп;
- в зависимости от группы и состояния процессора прерывание инициирует либо FIQ, либо IRQ запрос в ядра CPU.

3.2.7 Доверенная накристальная загрузочная память ROM

Доверенная накристальная загрузочная ROM объемом 64 Кбайта реализована внутри доверенного процессора RISC0 и доступна только для него по адресам, начиная с 0x1FC0_0000, т.е. вектора сброса для данного процессора.

3.2.8 Доверенная накристальная память RAM

Доверенная накристальная загрузочная RAM объемом 32 Кбайта реализована внутри доверенного процессора RISC0, в виде памяти его быстрого доступа CRAM, и доступна только для доверенных мастеров service_subs.

3.2.9 Доверенные устройства ввода-вывода и контроллеры памяти

Доверенный контур содержит следующие выделенные устройства ввода-вывода и контроллеры памяти:

- Многофункциональный контроллер MFBSP0 (I2S, SPI, GPIO) – аудио вывод, контроль клавиатуры и др. устройств ввода/вывода, контроллер внешней доверенной SPI Flash.
- Многофункциональный контроллер MFBSP1 (I2S, SPI, GPIO) – аудио вывод, контроль клавиатуры и др. устройств ввода/вывода, контроллер внешней доверенной SPI Flash.
- Контроллер Flash память QSPI0 с функцией прямого исполнения (XiP – eXecution in Place) – внешнее доверенное хранилище программного кода.

Данные контроллеры доступны для только для доверенных обращений.

3.2.10 Доверенное хранилище ключевой информации

Доверенное хранилище реализовано с помощью массива энергонезависимых fuse-ячеек, формирующих однократно программируемую память – OTP. Память доступна в виде набора регистров, доступных только по чтению и только для доверенных обращений. Список регистров приведен в Разделе 5.6.2

3.2.11 Безопасность доступа к памяти DDR

Доступ к DDR памяти осуществляется через два диапазона в физической карте памяти – DDR Low и DDR High. Диапазон DDR Low предназначен для использования доверенными средами исполнения и приложениями связного и доверенного контура. Диапазон разбит на четыре региона памяти по два на связной и доверенный контур:

Таблица 3.5. Диапазон памяти DDR Low

Стартовый адрес	Размер области	Описание	Подсистема, порт	Уровень доступа
0x0080000000	512 Мбайт	DDR Low Trusted 0	ddr	trusted, securable
0x00a0000000	512 Мбайт	DDR Low SDR 0	ddr	sdr, securable
0x00c0000000	512 Мбайт	DDR Low Trusted 1	ddr	trusted, securable
0x00e0000000	512 Мбайт	DDR Low SDR 1	ddr	sdr, securable

Таким образом, по умолчанию доступ в диапазон памяти DDR Low ограничен только для обращений с привилегиями trusted и sdr, т.е. для обращений из доверенного и связного контура соответственно. Запись и чтение с помощью обращений с более низкими правами будет приводить к error response на шине AXI.

Для регионов в DDR Low предусмотрена, однако, возможность понизить порог прав доступа обращений до secure с помощью регистра DDR_LOW_RANGE_SECURE_CTR (см Раздел 3.9.2.5).

Диапазон адресов DDR High является общим и с памятью в этом диапазоне по умолчанию могут работать мастера в любом из контуров микросхемы.

Также подсистема ddr содержит регистры (см. Таблица 3.6), с помощью которых можно ограничить доступ к определенному диапазону адресов DDR памяти микросхемы до уровня secure. Запись в такой диапазон будет не иметь действия, а чтение будет возвращать нули. Кроме того, можно настроить контроллеры DDRMC0 и DDRMC1 таким образом (см. Описание регистра POISONCFG, Раздел 24.6.108), чтобы попытка доступа в диапазон приводила либо к ошибке на шине AXI SLVERR, либо к генерации прерывания.

Таблица 3.6. Регистры контроля доступа в DDR

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
CTRSECUREREGION	Регистр управления доступом к выделенным адресным диапазонам памяти.	0x0	W/R	0x80
BASESECUREREGION0_LOW	База адресного диапазона 0. Младшая часть.	0x0	W/R	0x90
BASESECUREREGION0_HIGH	База адресного диапазона 0. Старшая часть	0x0	W/R	0x94
MASKSECUREREGION0_LOW	Маска адресного диапазона 0. Младшая часть.	0x0	W/R	0x98
MASKSECUREREGION0_HIGH	Маска адресного диапазона 0. Старшая часть.	0x0	W/R	0x9C
BASESECUREREGION1_LOW	База адресного диапазона 1. Младшая часть.	0x0	W/R	0xA0
BASESECUREREGION1_HIGH	База адресного диапазона 1. Старшая часть	0x0	W/R	0xA4

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
MASKSECUREREGION1_LOW	Маска адресного диапазона 1. Младшая часть.	0x0	W/R	0xA8
MASKSECUREREGION1_HIGH	Маска адресного диапазона 1. Старшая часть.	0x0	W/R	0xAC
BASESECUREREGION2_LOW	База адресного диапазона 2. Младшая часть.	0x0	W/R	0xB0
BASESECUREREGION2_HIGH	База адресного диапазона 2. Старшая часть	0x0	W/R	0xB4
MASKSECUREREGION2_LOW	Маска адресного диапазона 2. Младшая часть.	0x0	W/R	0xB8
MASKSECUREREGION2_HIGH	Маска адресного диапазона 2. Старшая часть.	0x0	W/R	0xBC
BASESECUREREGION3_LOW	База адресного диапазона 3. Младшая часть.	0x0	W/R	0xC0
BASESECUREREGION3_HIGH	База адресного диапазона 3. Старшая часть	0x0	W/R	0xC4
MASKSECUREREGION3_LOW	Маска адресного диапазона 3. Младшая часть.	0x0	W/R	0xC8
MASKSECUREREGION3_HIGH	Маска адресного диапазона 3. Старшая часть.	0x0	W/R	0xCC

Доступ в регистры из таблицы выше возможен только для доверенных (trusted) обращений.

3.2.12 Доверенный генератор случайных чисел RNG

Генератор случайных чисел RNG с собственным источником энтропии (seed) может генерировать поток данных, которые статистически эквивалентны равномерно распределенному потоку случайных данных. Генератор отвечает набору тестов NIST 800-22 и может генерировать случайные числа разрядностью 128 и 256 бит.

3.2.13 Доверенные энергонезависимые счетчики

Доверенные энергонезависимые счетчики являются обязательными компонентами TBSA и применяются для хранения версий прошивок доверенного ПО и других компонентов ПО, значимых для пользователя. Аппаратно счетчики реализованы в виде нескольких наборов бит в доверенном хранилище ключевой информации – OTP, при этом каждый бит отвечает за определенное значение счетчика. Инкремент производится последовательным прожигом нового бита в наборе. При достижении максимального значения счетчик не сбрасывается и остается константным. Реализовано два обязательных счетчика (Trusted Firmware Updates Counter TFUC и Non-trusted Firmware Updates Counter FUC) и несколько счетчиков версий для пользовательских компонентов ПО (SWIDx), а также версий ID для периферийных устройств, подключаемых к микросхеме.

3.2.14 Доверенный watchdog таймер

Доверенный контур содержит выделенный watchdog таймер WDT0, который защищает систему в процессе загрузки, например, при необходимости запуска драйверов для не доверенных компонентов. При выработке таймера поступает прерывание через контроллер

прерываний QLIC0, если ПО загрузки не может очистить прерывание и таймер завершает счет второй раз, генерируется глобальный аппаратный сброс микросхемы. Подробнее см. в Главе 38 «Контроллер watchdog timer (WDT)».

Примечание: watchdog таймер общего контура WDT1 не может инициировать аппаратный сброс микросхемы и реализован для применения ПО уровня EL2 в рамках архитектурных требований ARM Server Base System Architecture.

3.2.15 Доверенное управление мощностью и тактовой частотой

В состав доверенного контура входят корневые контроллеры управления мощностью (PMCTR) и тактовой частотой (CMCTR) микросхемы.

В функции контроллера PMCTR входит включение и выключение доменов питания микросхемы. Управляющие регистры контроллера PMCTR доступны только для доверенных обращений, соответственно весь функционал ПО микросхемы, связанный с включением или выключением доменов питания микросхемы, должен быть реализован только в рамках доверенного ПО. Этот функционал может быть доступен общему ПО в виде запросов к сервисам доверенного ПО, вызываемых через аппаратный фильтр обращений (см. Раздел 3.2.3).

В функции контроллера CMCTR входит управление корневой частью системы синхронизации микросхемы, т.е. управление формированием опорных тактовых частот для подсистем микросхемы от опорной тактовой частоты микросхемы. Управляющие регистры контроллера CMCTR доступны только для доверенных обращений. Кроме этого, в каждой подсистеме микросхемы есть собственный контроллер, управляющий формированием тактовых частот блоков подсистемы от опорной частоты подсистемы. По умолчанию, доступ к управляющим регистрам контроллеров частоты подсистем также доступен только для доверенных обращений, однако, существует механизм, позволяющий через доверенные регистры TRUSTEDTOSECURE и SDRTOSECURE разрешить доступ для обращений уровня secure.

3.2.16 Безопасность и отладка микросхемы

Существует два механизма отладки микросхемы:

Отладка микросхемы через выделенный JTAG порт доверенного процессора RISC0 и встроенный отладчик OnCD. При такой отладке все запросы отладчика имеют уровень доступа доверенный и имеют возможность обратиться ко всем ресурсам микросхемы без исключения, в том числе отладочным регистрам CPU, DSP, RISC1 и ресурсам доверенного контура. Данный механизм применяется только при производственном тестировании и отладке микросхемы, после этого доступ к микросхеме через выделенный порт JTAG запрещается пережиганием fuse-бита (BRINGUPDBG_DISABLE в регистре FUSE0).

Основной механизм с помощью системы отладки микросхемы DEBUG на базе технологии UltraSoC. В рамках архитектуры TBSA в микросхеме существуют несколько механизмов защиты доступа для элементов системы отладки микросхемы – Debug Protection Mechanisms (DPM). Они, и права доступа, которые они дают, перечислены в Таблица 3.7.

Таблица 3.7. Механизмы защиты доступа для системы отладки

DPM	Уровень отладочного доступа в состоянии Unlock или Default	Состояние Locked	Аппаратные сигналы / регистры для Unlock
DPM_TP	Доверенный (trusted) для Privileged Space и User Space в RISC0	есть	- / TP_DBGEN
DPM_SDR	Связной (sdr) для Privileged Space и User Space в RISC1, DSP0, DSP1.	есть	- / SDR_DBGEN
DPM_SP	Безопасный (secure) для Privileged и User Space в CPU	есть	CPU.DBGEN, / U_DBGEN CPU.NIDEN, / U_DBGEN CPU.SPIDEN, / SP_DBGEN CPU.SPINDEN / SP_DBGEN
DPM_SU	Безопасный (secure) для User Space в CPU	-	CPU.DBGEN, / U_DBGEN CPU.SUNIDEN, / U_DBGEN CPU.NIDEN / U_DBGEN
DPM_NSP	Общий (non-secure) Privileged и User Space в CPU	-	CPU.DBGEN / U_DBGEN CPU.NIDEN / U_DBGEN
DPM_NSU	Общий (non-secure) для User Space в CPU	-	self-hosted debug на CPU

Каждый DPM имеет три или четыре (см. Таблица 3.7) состояния, которые отражают возможности отладочного доступа к микросхеме. Состояния DPM показаны на Рисунок 3.2.

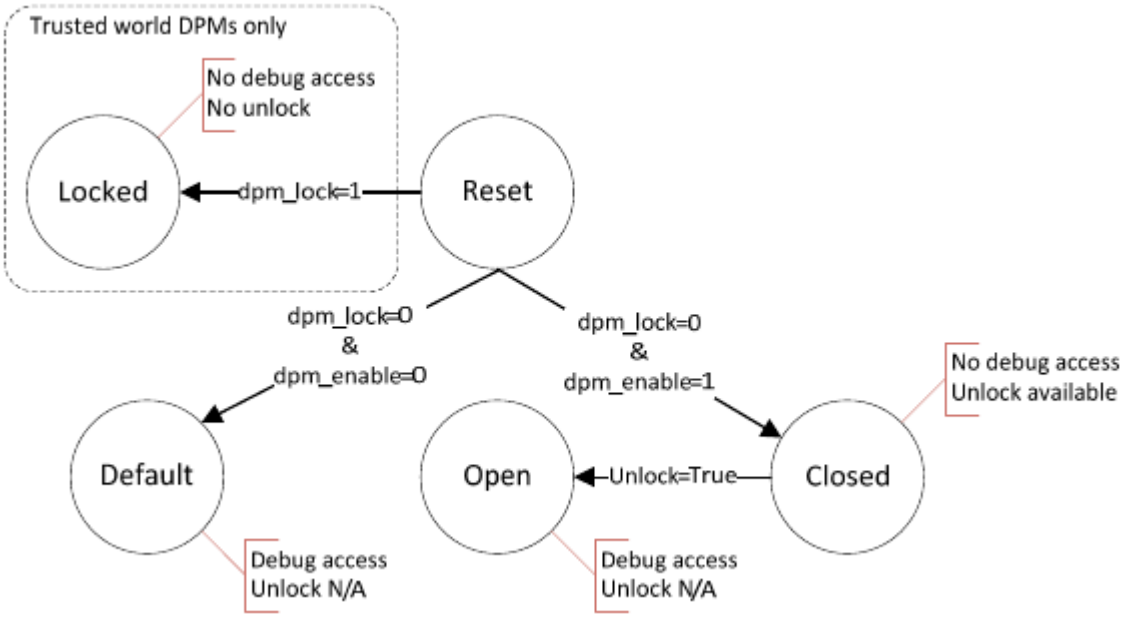


Рисунок 3.2. Состояния механизмов защиты отладочного доступа DPM

Начальное состояние DPM после сброса определяется набором fuse-битов DPM_LOCK и DPM_ENABLE. В микросхеме реализованы fuse-биты DPM_LOCK_TRUSTED для DPM_TP, DPM_LOCK_SDR для DPM_SDR, dpm_lock_secure для DPM_SP и общий для всех DPM бит DPM_ENABLE. См. регистр FUSE0.

Переход DPM из состояния Closed в состояние Open осуществляется операцией unlock, которая должна требовать токен безопасности, либо в виде простого пароля, либо в виде криптографически подписанного сертификата, это зависит от реализации в ПО отладки. После получения токена безопасности ПО отладки использует для операции unlock регистры _DBGEN, контролирующие аппаратные сигналы, разрешающие отладку (см. Таблица 3.7). Регистры _DBGEN доступны только для доверенных обращений, поэтому для системы отладки возможны два подхода к выполнению операции unlock:

- Во Flash памяти резервируется место под токены безопасности, и операция ulock выполняется во время процедуры доверенной загрузки в ПО загрузчика.
- Отладочное ПО обращается через UltraSoC к доверенному контуру с запросом на выполнение операции с помощью сервиса в Trusted Execution Environment.

Компоненты системы отладки DEBUG (UltraSoC) по умолчанию имеют общий уровень доступа (non-secure).

3.2.17 Загрузка микросхемы

Микросхема поддерживает механизм доверенной загрузки:

1) После подачи питания или сброса начинается выполнение доверенным ядром RISC0 программного кода первичного загрузчика из доверенной накристальной ROM
2) ПО первичного загрузчика начинает загрузку ПО вторичного загрузчика, хранящегося во внешнем ЗУ (QSPI0, QSPI1, UART0, SDMMC0, MFBSP0)
3) Если задан режим проверки подлинности (аутентификации) с публичным ключом, проверяется подлинность загружаемого ПО вторичного загрузчика.
4) Если задан режим шифрования, то загруженное ПО расшифровывается и проверяется его подлинность, в случае успешного завершения, управление передается ПО вторичного загрузчика
5) ПО вторичного загрузчика может выполнять следующие задачи:
    a. Настройка системы синхронизации микросхемы
    b. Настройка режима мощности микросхемы, выключение ненужных доменов питания
    c. Настройка признака доступа securable для областей микросхемы
    d. Выполнение операций unlock для DPM микросхемы
    e. Загрузка и аутентификация ПО загрузки для ядер CPU, RISC1, DSP0, DSP1

f. Старт ядер CPU, RISC1, DSP0, DSP1 программным снятием сигналов сброса, с использованием регистров _PPOLICY.

3.2.18 Замечания к аппаратной реализации функционала безопасности

- С использованием аппаратного фильтра доступа существенно сокращается уязвимый периметр, т.е. набор аппаратных ресурсов (регистры, память, выводы) потенциально доступных для атак.
- Есть возможность реализовать «классический» подход к TrustZone с отказом от использования аппаратного доверенного процессора и использованием только двух уровней доступа — secure, non-secure. В таком случае используется стандартный подход ARM с выполнением TEE и REE на CPU.
- Доверенный контур не поддерживает дешифрацию и шифрацию данных на лету, что в некоторых случаях не позволяет хранить доверенное исполняемое ПО во внешней памяти в шифрованном виде. Это потенциально делает возможным его перехват на плате через выводы QSPI0 и MFBSP.

3.3 Система коммутации микросхемы

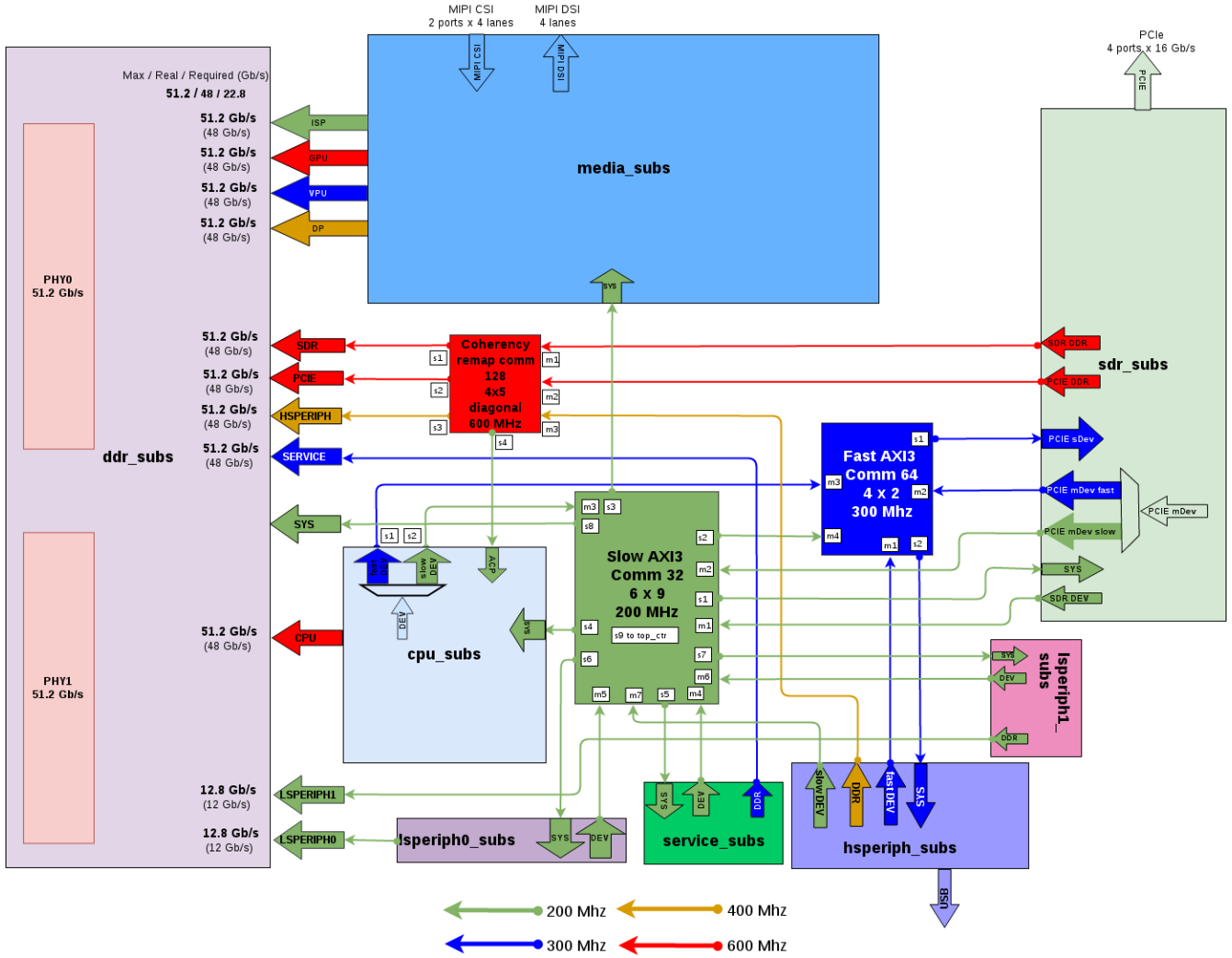


Рисунок 3.3. Структурная схема системы коммутации микросхемы

Системный интерконнект не допускает трансляцию транзакций, пересекающих адресную границу в 4Кбайта.

Таблица 3.8. Параметры интерфейсов коммутационной среды

Название подсистемы	Название интерфейса	Тип интерфейса	Ширина шины данных	Ширина ID	Максимальное число уникальных ID	Максимальное число транзакций на ID
CPU	ddr	AXI4 master	128	7	16	16
	slow_dev	AXI4 master	128	7	4	4
	fast_dev	AXI4 master	128	7	8	8
	sys	AXI4 slave	32	1	2	2
	acp	AXI4 slave	128	5	1	4
Service	ddr	AXI3 master	64	3	8	8
	dev	AXI3 master	64	3	2	4
	sys	AXI3 slave	32	1	2	4
Media	isp	AXI3 master	256	6	16	1
	gpu	AXI3 master	128	6	16	1
	vpu	AXI3 master	128	8	16	1
	dp	AXI4 master	64	11	16	1
	sys	AXI3 slave	32	1	2	2
SDR	sdr_ddr	AXI3 master	128	9	16	16
	sdr_dev	AXI3 master	128	9	8	8
	pcie_ddr	AXI3 master	128	8	16	1
	pcie_mdev_fast	AXI3 master	128	8	16	1
	pcie_mdev_slow	AXI3 master	128	8	16	1
	pcie_sdev	AXI3 slave	128	6	16	1
	sys	AXI3 slave	32	12	2	2
HSPeriph	ddr	AXI3 master	64	8	12	4
	fast_dev	AXI3 master	64	8	4	4
	slow_dev	AXI3 master	64	8	4	4
	sys	AXI3 slave	32	4	4	9
LSPeriph0	ddr	AXI3 master	32	1	2	2
	dev	AXI3 master	32	1	2	2
	sys	AXI3 slave	32	1	2	2
LSPeriph1	ddr	AXI3 master	32	1	2	2
	dev	AXI3 master	32	1	2	2
	sys	AXI3 slave	32	1	2	2
DDR	isp	AXI3 slave	256	4	16	1

	gpu	AXI3 slave	128	4	16	1
	vpu	AXI3 slave	128	4	16	1
	dp	AXI3 slave	64	4	16	1
	sdr	AXI3 slave	128	4	16	16
	pcie	AXI3 slave	128	4	16	1
	hsperiph	AXI3 slave	64	4	8	8
	lsperiph0	AXI3 slave	128	4	1	1
	lsperiph1	AXI3 slave	128	4	1	1
	cpu	AXI4 slave	128	4	16	16
	service	AXI3 slave	64	4	8	8
	sys	AXI3 slave	32	4	32	32

Таблица 3.9. Матрица коммутации 1

		Slave интерфейсы												
		подсистема	Ddr											cpu
Master интерфейсы	подсистема	интерфейс	isp	gpu	vpu	dp	sdr	pcie	hsperiph	service	cpu	lsperiph0	lsperiph1	acp
	media	isp	+											
		gpu		+										
		vpu			+									
		dp				+								
	sdr	sdr_ddr					+							+
		pcie_ddr						+						+
	hsperiph	ddr							+					+
	service	ddr								+				
	cpu	ddr									+			
	lsperiph0	ddr										+		
	lsperiph1	ddr											+	

Таблица 3.10. Матрица коммутации 2

		Slave интерфейсы										
		подсистема	Sdr		media	cpu	service	hsperiph	lsperiph0	lsperiph1	ddr	comm_ctr
Master интерфейсы	Подсистема	ингтерфейс	Sys	pcie_sdev	sys	sys	sys	sys	sys	sys	sys	sys
	sdr	pcie_mdev_fast						+				
		pcie_mdev_slow	+		+	+	+		+	+	+	
		sdr_dev		+	+	+	+	+	+	+	+	
	cpu	dev_fast		+				+				
		dev_slow	+		+	+	+		+	+	+	
	service	dev	+	+	+	+		+	+	+	+	+
	hsperiph	dev_fast		+								
	hsperiph	dev_slow	+		+	+	+		+	+	+	+
	lsperiph0	dev		+ (только pcie0 low и pcie1 low)				+	+	+		
	lsperiph1	dev		+ (только pcie0 low и pcie1 low)				+	+	+		

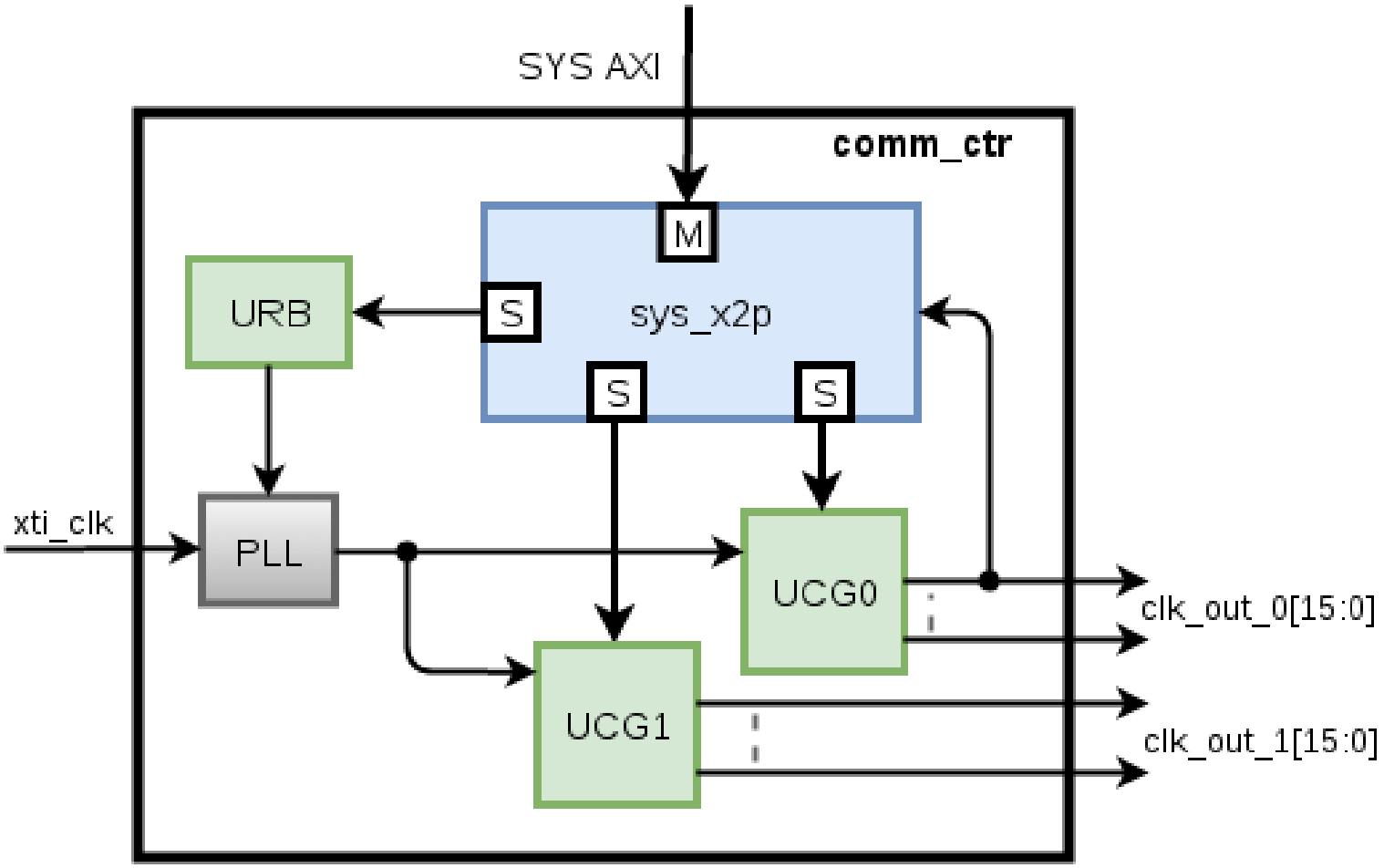


Рисунок 3.4. Структурная схема контроллера коммутационной среды comm_ctr

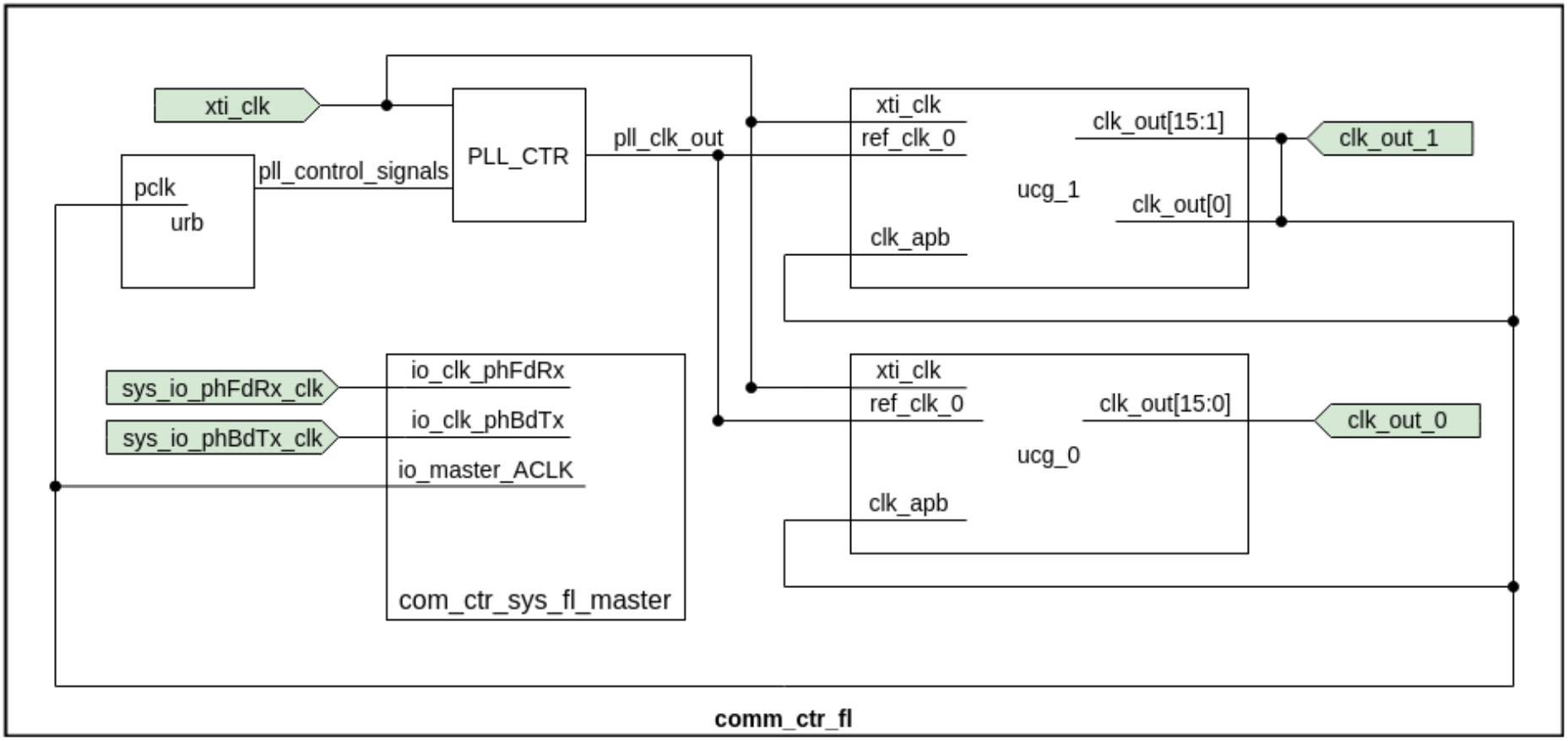


Рисунок 3.5. Схема формирования частот для подсистемы коммутации в comm_ctr

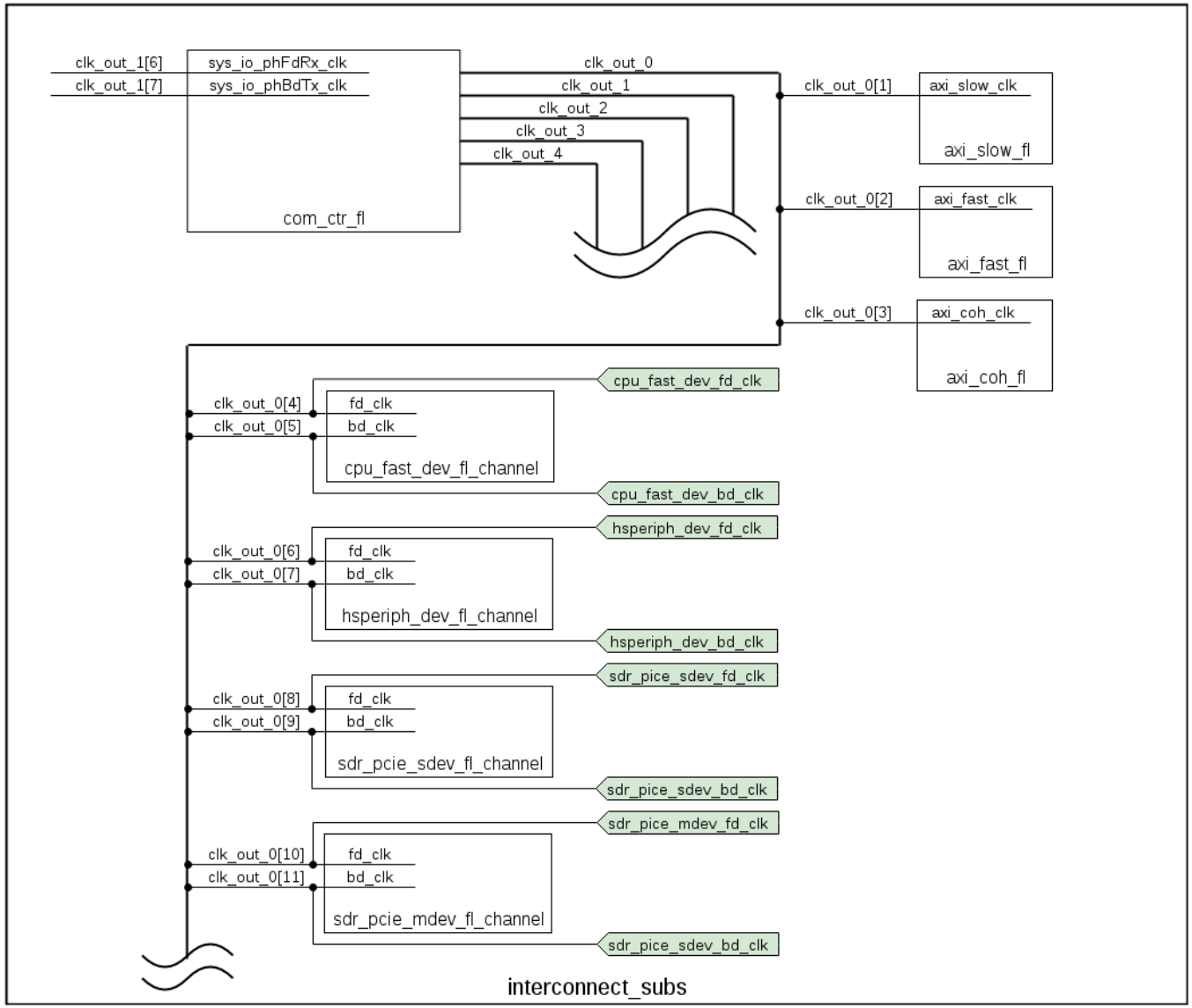


Рисунок 3.6. Схема распределения тактовых сигналов внутри подсистемы комутации

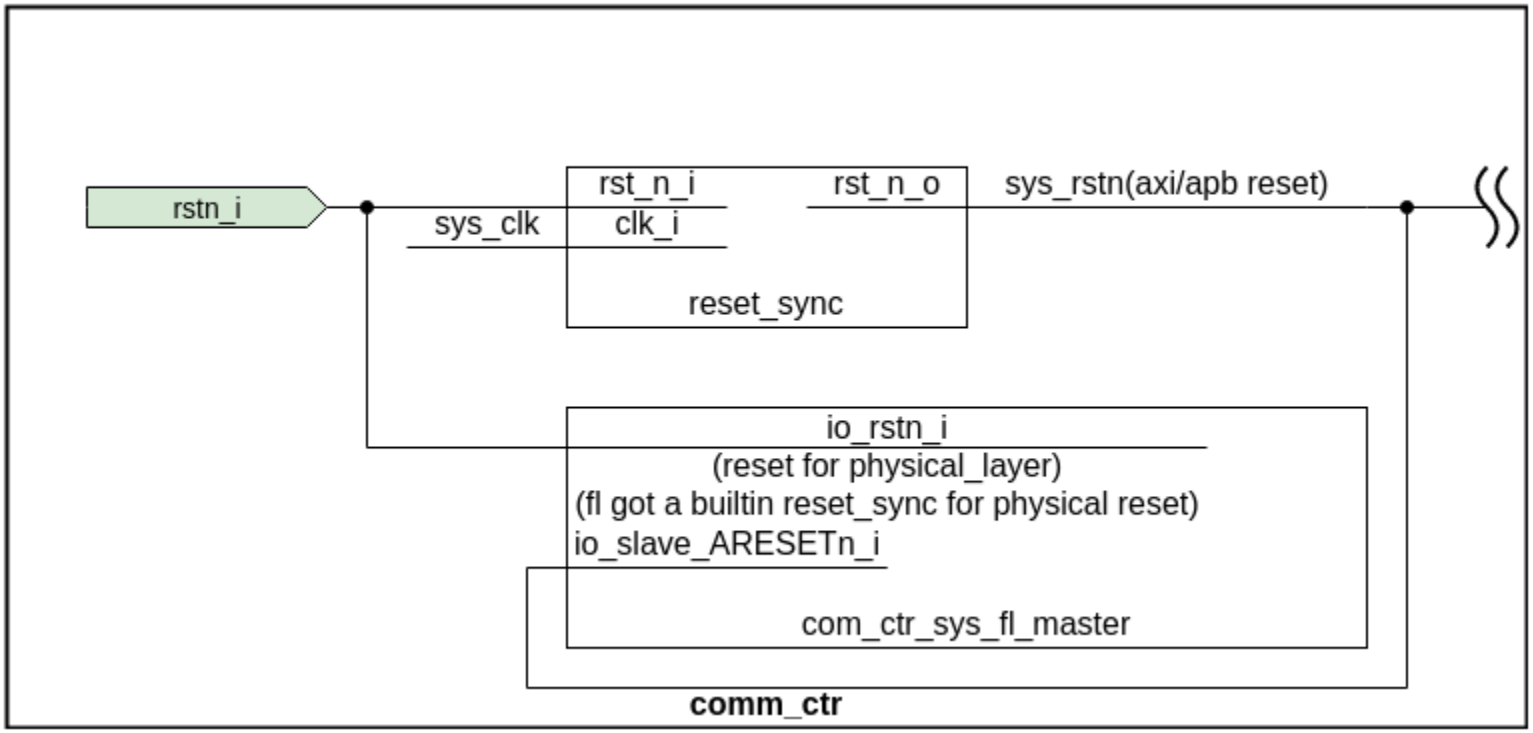


Рисунок 3.7. Схема сброса в comm_ctr

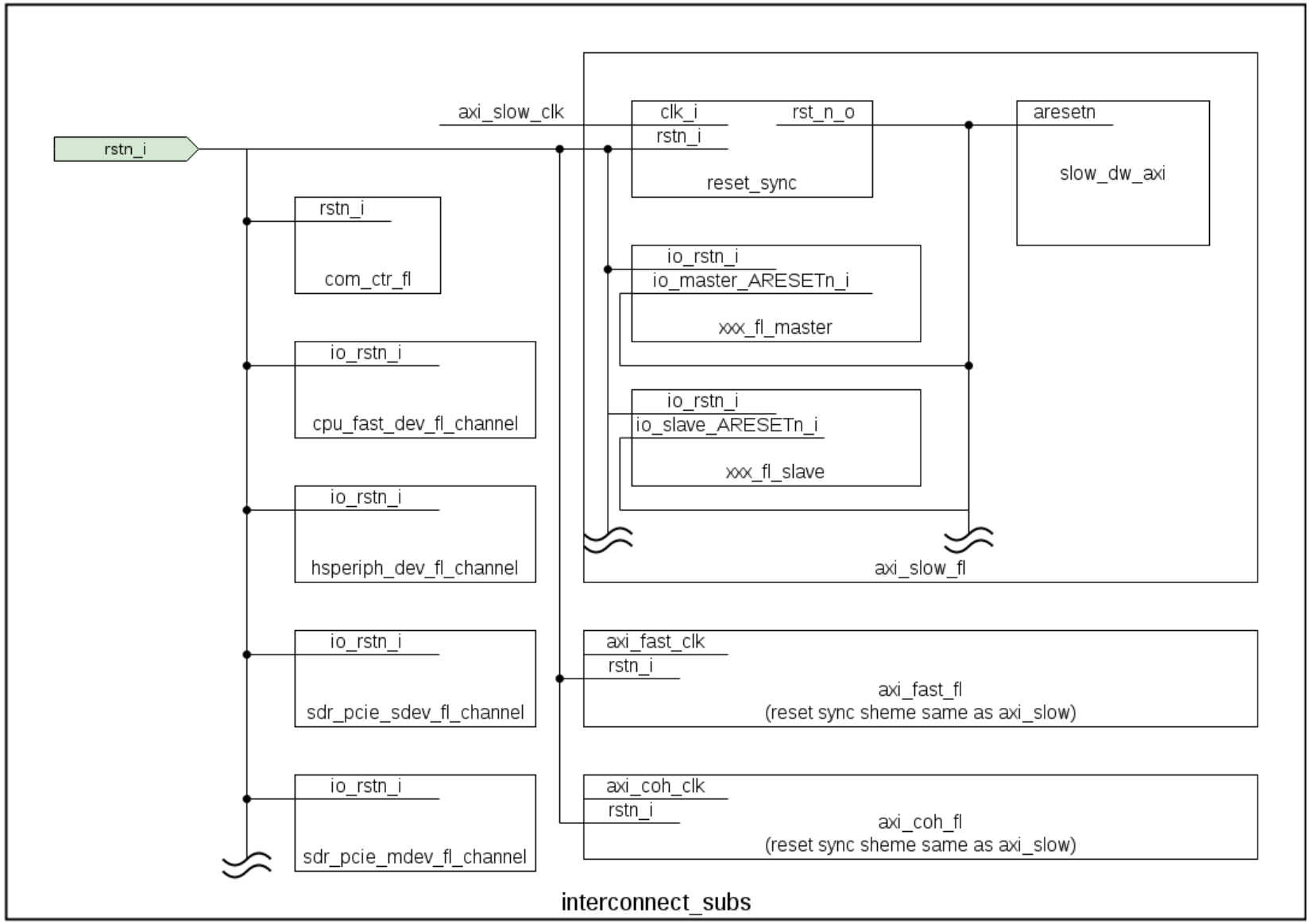


Рисунок 3.8. Схема сброса в interconnect_subs

Таблица 3.11. Источники тактовых сигналов в среде коммутации. Коэффициент деления UCG рассчитан для xti_clk = 27.456 Mhz и коэффициента умножения pllctr=44.

№	Источник	Канал	Частота	Кофициент деления UCG
1	clk_out_0[0]	ddr_dp	400	3
2	clk_out_0[1]	ddr_vpu	300	4
3	clk_out_0[2]	ddr_gpu	600	2
4	clk_out_0[3]	ddr_isp	200	6
5	clk_out_0[4]	ddr_cpu	600	2
6	clk_out_0[5]	cpu_acp	600	2
7	clk_out_0[6]	ddr_lsperiph0	200	6
8	clk_out_0[7]	axi_coh_comm	600	2
9	clk_out_1[0]	axi_slow_comm	200	6
10	clk_out_1[2]	axi_fast_comm	300	4
11	clk_out_1[4]	ddr_sdr_dsp	600	2
12	clk_out_1[5]	ddr_sdr_pcie	600	2
13	clk_out_1[6]	ddr_lsperiph1	200	6
14	clk_out_1[7]	ddr_service	300	4
15	clk_out_1[8]	ddr_hsperiph	400	3

3.4 Управление мощностью

3.4.1 Введение

В микросхеме вводится понятие домена питания. Домен питания – это часть логики микросхемы, отключение питания от которой не приводит к сбоям в других частях микросхемы, а включение возвращает в состояние по сбросу.

Домен питания может включать в себя логику, работающую на одной или нескольких тактовых частотах.

Домен питания может находиться в одном из двух состояний:

1. Включен – логика внутри домена питания работает в диапазоне рабочих частот, при этом все или некоторые тактовые частоты внутри домена могут быть отключены.

2. Выключен – отключено питание от всей логики внутри домена, включая память.

Концепция состояний домена питания покрывает весь спектр решений по уменьшению потребляемой мощности логики внутри домена:

- для снижения динамического потребления:
  - в состоянии «Включен» могут понижаться рабочие тактовые частоты, все или несколько тактовых частот могут быть отключены;
- для снижения статического потребления домен питания может быть временно или постоянно выключен.

3.4.2 Домены питания микросхемы

Домены питания микросхемы показаны на Рисунок 3.9.

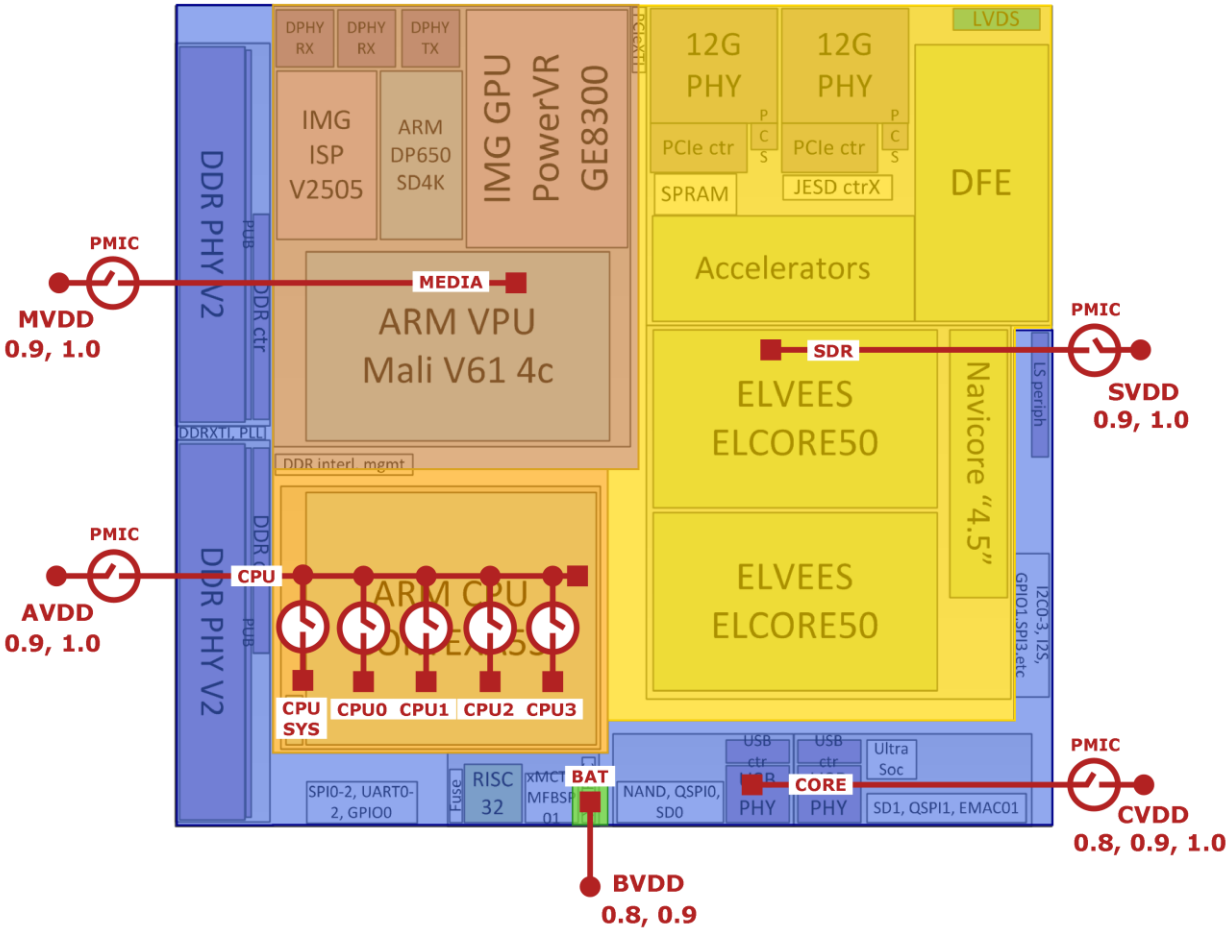


Рисунок 3.9. Домены питания микросхемы

1. Домен CORE включает подсистемы ddr_subs, hsperiph_subs, lsperiph0_subs, lsperiph1_subs, service_subs и коммутатор микросхемы верхнего уровня.

2. Домен CPU включает подсистему cpu_subs;

3. Домен MEDIA включает подсистему media_subs;

4. Домен SDR включает подсистему sdr_subs;

5. Домен BAT включает ядро таймера реального времени RTC (см. Главу 13), контактные площадки CORE_OFFREQ, XTI32K и NRST_PON;

6. Домен CPUSYS включает общую логику верхнего уровня кластера CortexA53 CPU, кэш L2, внутреннею коммутацию, интерфейсы и т.д.;

7. Домены CPU0, CPU1, CPU2, CPU3 включают логику процессорных ядер кластера CortexA53 CPU.

Состояния доменов питания управляются парами регистров *_PPOLICY, *_PSTATUS, имеющих состояния PP_ON и PP_OFF и соответствующих состоянию домена «Включен» и «Выключен».

Возможные режимы работы микросхемы и состояния ее доменов питания показаны в Таблица 3.12

Таблица 3.12. Режимы работы микросхемы

Режим	Домены питания	SDR	MEDIA	CORE	BAT	CPU	CPUSYS	CPU0	CPU1	CPU2	CPU3
Рабочий	Допустимые состояния домено	Вкл/ Выкл	Вкл/ Выкл	Вкл	Вкл	Вкл/ Выкл	как CPU/ Выкл	как CPU SYS/ Выкл	как CPU SYS/ Выкл	как CPU SYS/ Выкл	как CPU SYS/ Выкл
	Пример. Медиа приложение	Выкл	Вкл	Вкл	Вкл	Вкл	Вкл	Вкл	Выкл	Выкл	Выкл
	Пример. Связное приложение	Вкл	Выкл	Вкл	Вкл	Вкл	Вкл	Вкл	Вкл	Выкл	Выкл
	Пример. Контроллер	Выкл	Выкл	Вкл	Вкл	Вкл	Вкл	Вкл	Выкл	Выкл	Выкл
	Пример. Ожидание	Выкл	Выкл	Вкл	Вкл	Выкл	Выкл	Выкл	Выкл	Выкл	Выкл
Сон		Выкл	Выкл	Выкл	Вкл	Выкл	Выкл	Выкл	Выкл	Выкл	Выкл

Таблица 3.13. Режимы работы микросхемы

Режим	Напряжения	BAT	CORE	CPU	MEDIA	SDR
Рабочий	PM10	0.9	0.9	0.9	0.9	0.9
	PM11	0.9	0.9	0.9	0.9	off
	PM12	0.9	0.9	0.9	off	0.9
	PM13	0.9	0.9	0.9	off	off
	PM14	0.9	0.9	off	0.9	0.9
	PM15	0.9	0.9	off	0.9	off
	PM16	0.9	0.9	off	off	0.9
	PM17	0.9	0.9	off	off	off
OD	PM20	1.0	1.0	1.0	1.0	1.0
	PM21	1.0	1.0	1.0	1.0	off
	PM22	1.0	1.0	1.0	off	1.0
	PM23	1.0	1.0	1.0	off	off
Ожидание	PM30	0.8	0.8	off	off	off
Сон	PM31	0.8	off	off	off	off

Перед включением любого домена питания, кроме CORE, необходимо переключить UCG0 и UCG1 уровня top в bypass mode (UCG_BP_CTR_REG = 0xFFFF).

Домены CPU, MEDIA, SDR управляются соответствующими регистрами *_PPOLICY, *_PSTATUS в служебной подсистеме (см. Разделы 5.6.3 - 5.6.10). Состояние данных регистров связано с соответствующими управляющими выводами микросхемы *_OFFREQN и *_OFFACKN. Непосредственно отключение и включение питания домена должно выполняться внешней логикой на плате, исходя из состояния этих выводов.

Процедура выключения и включения домена при этом выглядит так как показано на Рисунок 3.10

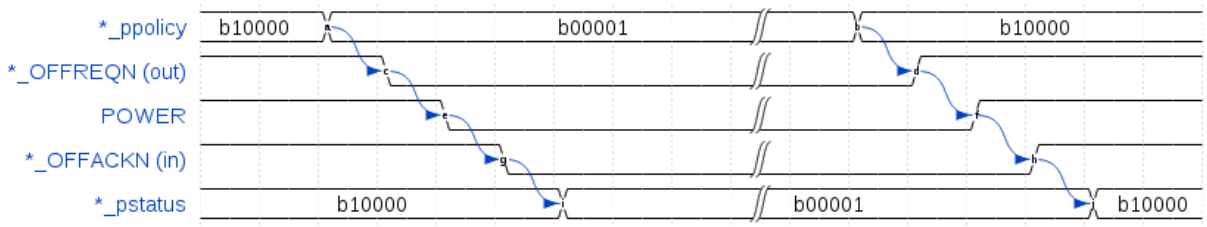


Рисунок 3.10. Процедура включения и выключения доменов CPU, MEDIA, SDR

Домены CPUSYS, CPU0, CPU1, CPU2, CPU3 управляются соответствующими регистрами *_PPOLICY, *_PSTATUS в подсистеме центрального процессора (см. Разделы 4.5.2 - 4.5.11).

3.5 Контроллеры прерываний в микросхеме

3.5.1 Обзор

В микросхеме присутствуют пять процессорных блоков, способных обрабатывать прерывания – CPU, RISC0, RISC1, DSP0, DSP1. В каждый из этих блоков прерывания поступают не напрямую, а через определенный котроллер прерываний. Для кластера CPU это контроллер прерываний GIC500, для ядра RISC0 – контроллер прерываний QLIC0, для ядер RISC1, DSP0, DSP1 – контроллер прерываний QLIC1. Подробное описание контроллеров см. в Главе 13 для GIC500 и Главе 19 для QLIC. Отображение прерываний микросхемы на контроллеры прерываний дано в таблице

3.5.2 Отображение прерываний

Таблица 3.14. Отображение прерываний в микросхеме

Номер прерывания / INTID	GIC500 PPI, SPI	Описание	QLIC0	Описание	QLIC1	Описание
0						
1			risc0_irq_dma		risc1_irq_dma	
2			risc0_irq_vmmu		risc1_irq_vmmu	
3			dsp0_event		sdr_irq_qlic0	Запрос на прерывание от QLIC0
4			dsp1_event			
5			cpu_event		cpu_event	
6			risc1_event		risc0_event	
7			axicore_dlock_irq | axisys_dlock_irq	прерывание от коммутатора подсистемы service		
8			mb0_rrq_risc0	событие от mailbox0 по запросу на чтение от risc0	mb1_rrq_risc0	событие от mailbox1 по запросу на чтение от risc0
9			mb0_rrq_risc1	событие от mailbox0 по запросу на чтение от risc1	mb1_rrq_risc1	событие от mailbox1 по запросу на чтение от risc1
10			mb0_rrq_dsp0	событие от mailbox0 по запросу на чтение от dsp0	mb1_rrq_dsp0	событие от mailbox1 по запросу на чтение от dsp0

Номер прерывания / INTID	GIC500 PPI, SPI	Описание	QLIC0	Описание	QLIC1	Описание
11			mb0_rrq_dsp1	событие от mailbox0 по запросу на чтение от dsp1	mb1_rrq_dsp1	событие от mailbox1 по запросу на чтение от dsp1
12			mb0_rrq_cpu0	событие от mailbox0 по запросу на чтение от cpu0	mb1_rrq_cpu0	событие от mailbox1 по запросу на чтение от cpu0
13			mb0_rrq_cpu1	событие от mailbox0 по запросу на чтение от cpu1	mb1_rrq_cpu1	событие от mailbox1 по запросу на чтение от cpu1
14			mb0_rrq_cpu2	событие от mailbox0 по запросу на чтение от cpu2	mb1_rrq_cpu2	событие от mailbox1 по запросу на чтение от cpu2
15			mb0_rrq_cpu3	событие от mailbox0 по запросу на чтение от cpu3	mb1_rrq_cpu3	событие от mailbox1 по запросу на чтение от cpu3
16	-		mfbsp0_rxirq	Формируется, если порт включен на прием данных (в одном из режимов), а число 64-х разрядных слов в буфере приёма больше, чем RLEV (RLEV устанавливается в регистре состояния приёмника RSR)	dsp0_irq: - TLB_DSPW - TLB_DSPR - TLB_DMAW - TLaB_DMAR	
17	-		mfbsp0_txirq	Формируется, если порт включен на передачу данных (в одном из режимов), а число 64-х разрядных слов, находящихся в буфере передачи меньше, либо равно TLEV (TLEV устанавливается в регистре состояния передатчика TSR)	dsp0_irq: - DMA0 - DMA1 - DMA2 - DMA3	
18	-		mfbsp0_srq	Запрос обслуживания от порта MFBSP0. Формируется, если порт выключен (LEN=0, SPI_I2S_EN=0), а на выводах LACK или LCLK присутствует сигнал высокого уровня	dsp0_irq: - PI - STP - SYSCALL - TAP - SE	
19	-		mfbsp1_rxirq	Формируется, если порт включен на прием данных (в одном из режимов), а число 64-	dsp0_irq: - dbBRK	

Номер прерывания / INTID	GIC500 PPI, SPI	Описание	QLIC0	Описание	QLIC1	Описание
				х разрядных слов в буфере приёма больше, чем RLEV (RLEV устанавливается в регистре состояния приёмника RSR)		
20	-		mfbsp1_txirq	Формируется, если порт включен на передачу данных (в одном из режимов), а число 64-х разрядных слов, находящихся в буфере передачи меньше, либо равно TLEV (TLEV устанавливается в регистре состояния передатчика TSR)	dsp0_irq: - BAD	
21	-		mfbsp1_srq	Запрос обслуживания от порта MFBSP0. Формируется, если порт выключен (LEN=0, SPI_I2S_EN=0), а на выводах LACK или LCLK присутствует сигнал высокого уровня	timers1_ch0_int	
22	cpu_ncommirq	событие, генерируемое Debug Communication Channel	mfbsp0_dmairq0	прерывание от канала DMA порта MFBSP0 при передаче данных	timers1_ch1_int	
23	cpu_npmuirq	событие, генерируемое Performance Counter;	mfbsp0_dmairq1	прерывание от канала DMA порта MFBSP0 при приеме данных	dsp1_irq: - TLB_DSPW - TLB_DSPR - TLB_DMAW - TLaB_DMAR	
24	cpu_ctiriqack	событие, генерируемое Cross Trigger Interface;	mfbsp1_dmairq0	прерывание от канала DMA порта MFBSP1 при передаче данных	dsp1_irq: - DMA0 - DMA1 - DMA2 - DMA3	
25	cpu_nvcpuntirq	конфигурируемое событие, генерируемое соответствующим virtual	mfbsp1_dmairq1	прерывание от канала DMA порта MFBSP1 при приеме данных	dsp1_irq: - PI - STP	

Номер прерывания / INTID	GIC500 PPI, SPI	Описание	QLIC0	Описание	QLIC1	Описание
		CPU interface для сигнала о ситуации, которая может требовать действий со стороны гипервизора;			- SYSCALL - TAP - SE	
26	cpu_ncnthpiirq	событие, генерируемое Non-secure EL2 physical timer	mfbsp0_canint0	прерывание 0 от CAN в MFBSP0	dsp1_irq: - dbBRK	
27	cpu_ncntvirq	событие, генерируемое virtual timer	mfbsp0_canint1	прерывание 1 от CAN в MFBSP0	dsp1_irq: - BAD	
28	qlic0_irqreq	запрос на обработку прерывания от qlic0;	mfbsp1_canint0	прерывание 0 от CAN в MFBSP1	timers1_ch2_int	
29	cpu_ncntpsirq	событие, генерируемое Secure EL1 physical timer	mfbsp1_canint1	прерывание 1 от CAN в MFBSP1	timers1_ch3_int	
30	cpu_ncntpnsirq	событие, генерируемое Non-secure EL1 physical timer;	pvt_ts_irq	прерывание от контроллера PVT по температурным датчикам	adfe_int	
31	qlic1_irqreq	запрос на обработку прерывания от qlic1;	pvt_vm_irq	прерывание от контроллера PVT по датчикам напряжения	adfe_rx_dma0_irq	
32	ls0_gpio0_int		rtc_it_event		adfe_rx_dma1_irq	
33	ls0_gpio1_int		rtc_alarm_event		adfe_rx_dma2_irq	
34	ls0_gpio2_int		otp_spi_intr	Прерывание от OTP_SPI	adfe_rx_dma3_irq	
35	ls0_gpio3_int		trng_irq	Прерывание от TRNG	adfe_mix_dma0_irq	
36	ls0_gpio4_int		wdt0_intr	Прерывание от WDT0	adfe_mix_dma1_irq	
37	ls0_gpio5_int		i2c4_ic_intr	Прерывание от I2C4	adfe_mix_dma2_irq	
38	ls0_gpio6_int		qspi0_intreq	Прерывание от QSPI0	adfe_mix_dma3_irq	
39	ls0_gpio7_int		-	-	ndfe_ddc0_dma0_irq	
40	qlic0 irqreq[0]		extint1	Прерывание с внешнего вывода EXTINT1	ndfe_ddc0_dma1_irq	
41	usb0_comb_irq		usb0_comb_irq	запрос на обработку прерывания от usb0	ndfe_ddc0_dma2_irq	
42	usb1_comb_irq		usb1_comb_irq	запрос на обработку прерывания от usb1	ndfe_ddc0_dma3_irq	
43	qlic1_irqreq		qlic1_irqreq	запрос на обработку прерывания от qlic1	ndfe_ddc1_dma0_irq	
44	isp_irq		isp_irq		ndfe_ddc1_dma1_irq	

Номер прерывания / INTID	GIC500 PPI, SPI	Описание	QLIC0	Описание	QLIC1	Описание
45	gpu_irq		gpu_irq		ndfe_ddc1_dma2_irq	
46	vpu_irq		vpu_irq		ndfe_ddc1_dma3_irq	
47	disp_de_irq		disp_de_irq		ndfe_ddc2_dma0_irq	
48	disp_se_irq		disp_se_irq		ndfe_ddc2_dma1_irq	
49	dsi_irq		dsi_irq		ndfe_ddc2_dma2_irq	
50	ddr0_dwc_ddrphy_int_n		ddr0_dwc_ddrphy_int_n		ndfe_ddc2_dma3_irq	
51	ddr0_dfi_alert_err_intr		ddr0_dfi_alert_err_intr		ndfe_ddc3_dma0_irq	
52	ddr0_derate_temp_limit_intr		ddr0_derate_temp_limit_intr		ndfe_ddc3_dma1_irq	
53	ddr0_derate_temp_limit_intr_fault		ddr0_derate_temp_limit_intr_fault		ndfe_ddc3_dma2_irq	
54	ddr1_dwc_ddrphy_int_n		ddr1_dwc_ddrphy_int_n		ndfe_ddc3_dma3_irq	
55	ddr1_dfi_alert_err_intr		ddr1_dfi_alert_err_intr		ndfe_adcbp_dma_irq	
56	ddr1_derate_temp_limit_intr		ddr1_derate_temp_limit_intr		ndfe_duc_dma0_irq	
57	ddr1_derate_temp_limit_intr_fault		ddr1_derate_temp_limit_intr_fault		ndfe_duc_dma1_irq	
58	ddr0_ecc_corrected_err_intr		ddr0_ecc_corrected_err_intr		ndfe_duc_dma2_irq	
59	ddr0_ecc_corrected_err_intr_fault		ddr0_ecc_corrected_err_intr_fault		ndfe_duc_dma3_irq	
60	ddr0_ecc_uncorrected_err_intr		ddr0_ecc_uncorrected_err_intr		ndfe_cmd_dma0_irq	
61	ddr0_ecc_uncorrected_err_intr_fault		ddr0_ecc_uncorrected_err_intr_fault		ndfe_cmd_dma1_irq	
62	ddr0_ecc_ap_err_intr		ddr0_ecc_ap_err_intr		ndfe_cmd_dma2_irq	
63	ddr0_ecc_ap_err_intr_fault		ddr0_ecc_ap_err_intr_fault		ndfe_cmd_dma3_irq	
64	ddr1_ecc_corrected_err_intr		ddr1_ecc_corrected_err_intr		ndfe_irq0	

Номер прерывания / INTID	GIC500 PPI, SPI	Описание	QLIC0	Описание	QLIC1	Описание
65	ddr1_ecc_corrected_err_intr_fault		ddr1_ecc_corrected_err_intr_fault		ndfe_irq1	
66	ddr1_ecc_uncorrected_err_intr		ddr1_ecc_uncorrected_err_intr		ndfe_irq2	
67	ddr1_ecc_uncorrected_err_intr_fault		ddr1_ecc_uncorrected_err_intr_fault		ndfe_irq3	
68	ddr1_ecc_ap_err_intr		ddr1_ecc_ap_err_intr		ndfe_drq0	
69	ddr1_ecc_ap_err_intr_fault		ddr1_ecc_ap_err_intr_fault		ndfe_drq1	
70	-		ddr_poison_intr		ndfe_drq2	
71	irq_emac0_comb		irq_emac0_comb		ndfe_drq3	
72	irq_emac0_wkup		irq_emac0_wkup		ndfe_drq4	
73	irq_emac1_comb		irq_emac1_comb		ndfe_drq5	
74	irq_emac1_wku		irq_emac1_wkup		ndfe_drq6	
75	irq_nfc_comb		irq_nfc_comb		ndfe_drq7	
76	irq_pdma2_comb		irq_pdma2_comb		ndfe_drq8	
77	irq_qspi_comb		irq_qspi_comb		ndfe_drq9	
78	irq_sdmmc0_comb		irq_sdmmc0_comb		ndfe_drq10	
79	irq_sdmmc0_wkup		irq_sdmmc0_wkup		ndfe_drq11	
80	irq_sdmmc1_comb		irq_sdmmc1_comb		ndfe_drq12	
81	irq_sdmmc1_wkup		irq_sdmmc1_wkup		ndfe_drq13	
82	-		ust_dbg_comb		ndfe_drq14	
83	interconnect_dlk_event		interconnect_dlk_event		ndfe_drq15	
84	pdma0_int_comb		pdma0_int_comb		ndfe_drq16	
85	uart3_intr		uart3_intr		ndfe_drq17	
86	uart1_intr		uart1_intr		ndfe_drq18	
87	uart2_intr		uart2_intr		ndfe_drq19	
88	ssi0_intr		ssi0_intr		ndfe_drq20	
89	i2c0_intr		i2c0_intr		jesd0_rx_interrupt	
90	ls1_gpio0_int		ls1_gpio0_int		jesd0_tx_interrupt	
91	ls1_gpio1_int		ls1_gpio1_int		jesd1_rx_interrupt	
92	ls1_gpio2_int		ls1_gpio2_int		jesd1_tx_interrupt	
93	ls1_gpio3_int		ls1_gpio3_int		gnss_irq	

Номер прерывания / INTID	GIC500 PPI, SPI	Описание	QLIC0	Описание	QLIC1	Описание
94	ls1_gpio4_int		ls1_gpio4_int		vitdec_irq	
95	ls1_gpio5_int		ls1_gpio5_int		convenc_irq	
96	ls1_gpio6_int		ls1_gpio6_int		tdc_irq	
97	ls1_gpio7_int		ls1_gpio7_int		tec_irq	
98	timers0_ch0_int		timers0_ch0_int		fft_irq	
99	timers0_ch1_int		timers0_ch1_int		mpsm_irq	
100	timers0_ch2_int		timers0_ch2_int		dma0_irq0	
101	timers0_ch3_int		timers0_ch3_int		dma0_irq1	
102	timers0_ch4_int		timers0_ch4_int		dma0_irq2	
103	timers0_ch5_int		timers0_ch5_int		dma0_irq3	
104	timers0_ch6_int		timers0_ch6_int		dma0_irq4	
105	timers0_ch7_int		timers0_ch7_int		dma0_irq5	
106	i2c1_ic_intr		i2c1_ic_intr		dma0_irq6	
107	i2c2_ic_intr		i2c2_ic_intr		dma0_irq7	
108	i2c3_ic_intr		i2c3_ic_intr		dma1_irq0	
109	ssi1_ssi_intr		ssi1_ssi_intr		dma1_irq1	
110	i2s0_intr		i2s0_intr		dma1_irq2	
111	uart0_intr		uart0_intr		dma1_irq3	
112	pwm_int		pwm_int		dma1_irq4	
113	pmwtu_int		pmwtu_int		dma1_irq5	
114	pdma1_int_comb		pdma1_int_comb		dma1_irq6	
115	wdt1_intr		wdt1_intr		dma1_irq7	
116	pcie0_system_irq		pcie0_system_irq		pcie0_system_irq	
117	pcie0_error_irq		pcie0_error_irq		pcie0_error_irq	
118	pcie0_pme_irq		pcie0_pme_irq		pcie0_pme_irq	
119	pcie0_edma_irq		pcie0_edma_irq		pcie0_edma_irq	
120	pcie0_intx_irq		pcie0_intx_irq		pcie0_intx_irq	
121	pcie0_msi_irq		pcie0_msi_irq		pcie0_msi_irq	
122	pcie1_system_irq		pcie1_system_irq		pcie1_system_irq	
123	pcie1_error_irq		pcie1_error_irq		pcie1_error_irq	
124	pcie1_pme_irq		pcie1_pme_irq		pcie1_pme_irq	
125	pcie1_edma_irq		pcie1_edma_irq		pcie1_edma_irq	
126	pcie1_intx_irq		pcie1_intx_irq		pcie1_intx_irq	
127	pcie1_msi_irq		pcie1_msi_irq		pcie1_msi_irq	

3.5.3 Проброс прерываний между контроллерами в микросхеме

В ряде случаев существует возможность проброса прерываний от одного контроллера микросхемы к другому. Для этого вторичный контроллер назначается целью нового (промежуточного) прерывания для первичного контроллера. В таблице показаны возможности для проброса прерываний.

Таблица 3.15. Проброс прерываний между контроллерами

Первичный контроллер	Цель для промежуточного прерывания, номер цели	Вторичный контроллер	Прерывание во вторичном контроллере, номер прерывания
QLIC0	qlic0_irqreq[0], 4	GIC500	qlic0_irqreq[0] PPI28 CPU0, 28 и SPI40
	qlic0_irqreq[1], 5		qlic0_irqreq[1] PPI28 CPU1, 28
	qlic0_irqreq[2], 6		qlic0_irqreq[2] PPI28 CPU2, 28
	qlic0_irqreq[3], 7		qlic0_irqreq[3] PPI28 CPU3, 28
QLIC0	qlic0_irqreq[4], 8	QLIC1	sdr_irq_qlic0, 3
QLIC1	sdr_irq_gic, 16	GIC500	qlic1_irqreq[0] PPI31 CPU0, 31 и SPI43
			qlic1_irqreq[0] PPI31 CPU1, 31
			qlic1_irqreq[0] PPI31 CPU2, 31
			qlic1_irqreq[0] PPI31 CPU3, 31
QLIC1	sdr_irq_qlic0, 17	QLIC0	qlic1_irqreq, 43

3.6 Логика сброса микросхемы и включение ее питания

Базовой подсистемой, ответственной за логику за логику сброса микросхемы является служебная. Логика сброса описана в Главе 5.3.

Не существует никаких принципиальных ограничений на порядок подачи различных групп питания на микросхему. Однако, следует иметь в виду, что если напряжение питания входных/выходных драйверов подано до момента подачи напряжение ядра, то в таком случае могут наблюдаться значительные токи по цепи питания драйверов между этими двумя моментами, которые обусловлены отсутствием валидных сигналов управления для драйверов в этот промежуток. В тоже время, если напряжение питания ядра подано до напряжения питания входов/выходов, то ток по цепи питания ядра может незначительно увеличиться относительно нормального.

Таким образом, общей рекомендацией будет подавать напряжения ядра и входов/выходов одновременно, но допустимо, чтобы питание ядра подавалось за очень малое время до питания входов/выводов. Если подача питания входов/выходов возможна только до момента включения питания ядра, то рекомендуется разносить эти моменты на время не более 100 мс, для ограничения возможных токов по цепи питания входов/выходов.

Никаких зависимостей во времени подачи питания между различными группами питания входов/выходов не существует. Наклон фронта подачи питания на любую линию питания микросхемы должен быть меньше 5 мВ/мкс (подробнее см. в Разделе 50.1).

3.7 Счетчики и таймеры в микросхеме

3.7.1 Счетчик XTICNT

64-разрядный счетчик XTICNT, работающий на частоте XTI, является источником общей временной метки для подсистем. К нему подключены входы CNTVALUEB CPU и soc_timer GPU. Счетчик является бинарным, однако переводится в код грея для передачи в подсистемы. Счетчик реализован в подсистеме secure_subs и доступен как для доверенных (см. регистр TSCOUNT_HIGH), так и для secure обращений (см. регистр TSCOUNT_LOW). Для счетчика предусмотрен механизм сохранения и восстановления метки времени, во время выключения микросхемы, с питанием домена BAT от батареи. Механизм опирается на таймер реального времени RTC, работающий на частоте XTI_32K (подробнее в Главе 40 «Таймер реального времени (RTC)»).

3.8 Логика межпроцессорного взаимодействия в микросхеме

3.8.1 События

Процессорные ядра RISC0, RISC1, DSP0, DSP1 и ядра в кластере CPU имеют аппаратные сигналы посылки и приема событий, связанные с выполнением соответствующих инструкций, аналогичных инструкциям SEV и WFE архитектуры ARMv7. В микросхеме эти сигналы соединены по принципу каждый с каждым, таким образом, например, DSP0 ядро может, выполнив инструкцию SEV, вывести из состояния WFE CPU. В подсистемах media, cpu существуют регистры (CPU_EVENTIMASK) маскирующие события для того или иного ядра. Коммутация событий в микросхеме показана в таблице Таблица 3.16.

Таблица 3.16. Коммутация событий в микросхеме

Источник	Абоненты
SERVICE.RISC0.CSR.VCU.EVENTOUT	QLIC1.INT6, CPU.EVENTI, DSP0.EVENTS_IN[9] DSP1.EVENTS_IN[9]
SDR.RISC1.CSR.VCU.EVENTOUT	QLIC0.INT6, CPU.EVENTI, DSP0.EVENTS_IN[8] DSP1.EVENTS_IN[8]
DSP0.EVENTS_OUT[0]	QLIC0.INT3, CPU.EVENTI, DSP0.EVENTS_IN[4] DSP1.EVENTS_IN[4]
DSP0.EVENTS_OUT[3:1]	DSP0.EVENTS_IN[7:5] DSP1.EVENTS_IN[7:5]
DSP1.EVENTS_OUT[0]	QLIC0.INT4 CPU.EVENTI, DSP0.EVENTS_IN[0] DSP1.EVENTS_IN[0]
DSP1.EVENTS_OUT[3:1]	DSP0.EVENTS_IN[3:1] DSP1.EVENTS_IN[3:1]
CPU.EVENTO	QLIC0.INT5

	QLIC1.INT5 DSP0.EVENTS_IN[10] DSP1.EVENTS_IN[10]

3.9 Системные настройки микросхемы

3.9.1 Адресное пространство регистров уровня top.

Таблица 3.17. Адресное пространство регистров уровня top

Смещение	Конечный адрес	Описание
0x0000	0x0FFF	Регистры URB
0x1000	0x1FFF	Регистры UCG0
0x2000	0x2FFF	Регистры UCG1

3.9.2 Регистры URB

Таблица 3.18. Регистры URB

Условное Обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
PLLCFG	Регистр конфигурации PLL	0x0	W/R	0x00
PLLDIAG	Регистр управления коэффициентом умножения PLL подсистемы.	0x0	W/R	0x04
NONSECURETOSECURE	Регистр изменения уровня доступа для областей памяти микросхемы с non-secure на secure	0x10	W/R	0x08
SDRTOSECURE	Регистр изменения уровня доступа для областей памяти микросхемы с sdr на secure	0x1FF	W/R	0x0C
TRUSTEDTOSECURE	Регистр изменения уровня доступа для областей памяти микросхемы с trusted на secure	0x3FFF	W/R	0x10
DDR_LOW_RANGE_SECURE_CTR	Регистр управления доступом для области памяти ddr low.	0xF	W/R	0x14
TRUSTED_ID_0	ID 0 для транзакция политики trusted	0x1FFE0C00	W/R	0x20
TRUSTED_ID_1	ID 1 для транзакция политики trusted	0x1FFE0C02	W/R	0x24
TRUSTED_ID_2	ID 2 для транзакция политики trusted	0x1FFE0C04	W/R	0x28
TRUSTED_ID_3	ID 3 для транзакция политики trusted	0x1FFE0C06	W/R	0x2C
TRUSTED_ID_4	ID 4 для транзакция политики trusted	0x0000FFFF	W/R	0x30
TRUSTED_ID_5	ID 5 для транзакция политики trusted	0x0000FFFF	W/R	0x34
TRUSTED_ID_6	ID 6 для транзакция политики trusted	0x0000FFFF	W/R	0x38

Условное Обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
TRUSTED_ID_7	ID 7 для транзакция политики trusted	0x0000FFFF	W/R	0x3C
SDR_ID_0	ID 0 для транзакция политики sdr	0x1EF00000	W/R	0x40
SDR_ID_1	ID 1 для транзакция политики sdr	0x1FCE0200	W/R	0x44
SDR_ID_2	ID 2 для транзакция политики sdr	0x1FEE0240	W/R	0x48
SDR_ID_3	ID 3 для транзакция политики sdr	0x1FEC0260	W/R	0x4C
SDR_ID_4	ID 4 для транзакция политики sdr	0x1FF00280	W/R	0x50
SDR_ID_5	ID 5 для транзакция политики sdr	0x1FF80290	W/R	0x54
SDR_ID_6	ID 6 для транзакция политики sdr	0x1FCE0280	W/R	0x58
SDR_ID_7	ID 7 для транзакция политики sdr	0x1FEE02C	W/R	0x5C
SLOW_AXI_DLOCK	Статус прерывания slow_axi_dlock	0x0	R	0x60
FAST_AXI_DLOCK	Статус прерывания fast_axi_dlock	0x0	R	0x64
COH_AXI_DLOCK	Статус прерывания coh_axi_dlock	0x0	R	0x68
COH_REMAP	Регистр перенаправления когерентных потоков данных.	0x0	W/R	0x6C
CHIP_VERSION	Регистр версии топа микросхемы.		W/R	0x70
CPU_SUBS_VERSION	Регистр версии cpu_subs.		W/R	0x74
DDR_SUBS_VERSION	Регистр версии ddr_subs		W/R	0x78
INTERCONNECT_SUBS_VERSION	Регистр версии interconnect_subs		W/R	0x7C
SDR_SUBS_VERSION	Регистр версии sdr_subs		W/R	0x80
MEDIA_SUBS_VERSION	Регистр версии media_subs		W/R	0x84
SERVICE_SUBS_VERSION	Регистр версии service_subs		W/R	0x88
HSPERIPH_SUBS_VERSION	Регистр версии hsperiph_subs		W/R	0x8C
LSPERIPH0_SUBS_VERSION	Регистр версии lsperiph0_subs		W/R	0x90
LSPERIPH1_SUBS_VERSION	Регистр версии lsperiph1_subs		W/R	0x94

3.9.2.1 Регистр PLLCFG

Регистр конфигурации PLL.

Формат регистра PLLCFG приведен в следующей таблице:

Таблица 3.19. Поля регистра PLLCFG

Диапазон	Название	Описание	Сброс
7:0	sel	Определяет выходную частоту и режим работы PLL: 0x0 - опорная частота ref_clk, PLL выключена; 0x1 - ref_clk * 2; 0x2 - ref_clk * 3; 0x3 - ref_clk * 4; ... - ...; 0x73 - ref_clk * 116; 0x74 - ref_clk * 116; ... - ...; 0xFF - ref_clk * 116; При sel > 0 и man = 1, конфигурация pll определяется полями *_man данного регистра	
8:8	-	Резерв	0
9:9	man	При бите заданном в 1 и sel > 0 конфигурация pll определяется полям *_man данного регистра. Выходная частота определяется по формуле ref_clk * (nf_man + 1) / (nr_man + 1) / (od_man + 1). Значение выходной частоты не должно превышать 3600 МГц. Принцип расчета: Fout = VCO / OD, где VCO = xtal / NR * NF Ограничения: 720 МГц <= VCO <= 3600 МГц NR должен находится в диапазоне [1..64]. Рекомендуемое значение – 1 (для минимизации джиттера). NF должен находиться в диапазоне [1..4096]. OD должен быть четным числом или единицей и находиться в диапазоне [1..16]. Увеличение значения OD способствует снижению уровня джиттера. При записи в регистр значения NR, NF и OD неоходимо уменьшить на единицу. Например, для 594 МГц походят коэффициенты NR=1, NF=132, OD=6. В регистр записываются значения nr_man=0, nf_man=131, od_man=5.	0
13:10	od_man	Коэффициент od настройки pll	0
26:14	nf_man	Коэффициент nf настройки pll	0
30:27	nr_man	Коэффициент nr настройки pll	
31	lock	Бит указывает окончание настройки PLL на новую заданную частоту работы. При начале процедуры настройки либо при выключенной PLL сбрасывается в 0.	0

3.9.2.2 Регистр PLLDIAG

Регистр диагностики PLL.

Формат регистра PLLDIAG приведен в следующей таблице:

Таблица 3.20. Поля регистра PLLDIAG

Диапазон	Название	Описание	Сброс
0:0	test	Диагностический вход pll - bypass опорной частоты pll на выход при 1	0
1:1	ensat	Диагностический вход pll - разрешает насыщение, в нормальном режиме - 1	1
2:2	fasten	Диагностический вход pll - разрешает механизм fast lock	0
3:3	rfslip	Диагностический выход pll - регистрирует ситуацию referency cycle slip	0
4:4	fbslip	Диагностический выход pll - регистрирует ситуацию feedback cycle slip	0
31:5	-	Резерв	

3.9.2.3 Регистр NONESECURETOSECURE

Регистр изменения уровня доступа для областей памяти микросхемы с non-secure на secure.

Формат регистра NONESECURETOSECURE приведен в Таблица 3.21.

Таблица 3.21. Формат регистра NONESECURETOSECURE

Номер бита	Условное обозначение	Назначение
0	hsperiph_subs_sys_secure	При установке в 1 уровень доступа следующих областей изменяется на secure: 1. SPRAM0. 2. Регистры блоков подсистемы hsperiph_subs. 3. QSPI1 подсистемы hsperiph_subs. Также изменяется на secure уровень доступа всех мастер – блоков подсистемы hsperiph_subs.
1	cpu_subs_gic500_secure	При установке в 1 уровень доступа для области блока GIC500 подсистемы cpu_subs изменяется на secure.
2	media_subs_sys_secure	При установке в 1 уровень доступа для области регистров блоков media_subs изменяется на secure.
3	media_subs_dp0_secure	При установке в 1 уровень доступа для non-secure области регистров блока dp media_subs изменяется на secure.
4	media_subs_dp1_secure	При установке в 0 уровень доступа для secure области регистров блока dp media_subs изменяется на secure.
5	lsperiph0_subs_sys_secure	При установке в 1 уровень доступа для области регистров блоков подсистемы lsperiph0_subs изменяется на secure. Также изменяется на secure уровень доступа pdma0.
6	lsperiph1_subs_sys_secure	При установке в 1 уровень доступа для области регистров блоков подсистемы lsperiph1_subs изменяется на secure. Также изменяется на secure уровень доступа pdma1.
7	sdr_subs_pcie_low_secure	При установке в 1 уровень доступа для области PCIe Low изменяется на secure.
31:6	-	Резерв

3.9.2.4 Регистр SDRTOSECURE

Регистр изменения уровня доступа для областей памяти микросхемы с sdr на secure.

Формат регистра SDRTOSECURE приведен в Таблица 3.22.

Таблица 3.22. Формат регистра SDRTOSECURE

Номер бита	Условное обозначение	Назначение
0	sdr_subs_sys0_sdrtrusted	При установке в 0 область доступна для secure обращений.
1	sdr_subs_phypcie0_sdrtrusted	При установке в 0 область доступна для secure обращений.
2	sdr_subs_sys1_sdrtrusted	При установке в 0 область доступна для secure обращений.
3	sdr_subs_phypcie1_sdrtrusted	При установке в 0 область доступна для secure обращений.
4	sdr_subs_mailbox1_sf_sdrtrusted	При установке в 0 область доступна для secure обращений.
5	sdr_subs_sys2_sdrtrusted	При установке в 0 область доступна для secure обращений.
6	sdr_subs_ctrpcie0_sdrtrusted	При установке в 0 область доступна для secure обращений.
7	sdr_subs_ctrpcie1_sdrtrusted	При установке в 0 область доступна для secure обращений.
8	sdr_subs_mem_sdrtrusted	При установке в 0 область доступна для secure обращений.
31:9	-	Резерв

3.9.2.5 Регистр DDR_LOW_RANGE_SECURE_CTR

Регистр изменения уровня доступа к диапазону памяти ddr_low микросхемы.

Формат регистра DDR_LOW_RANGE_SECURE_CTR приведен в таблице.

Таблица 3.23. Формат регистра DDR_LOW_RANGE_SECURE_CTR

Номер бита	Условное обозначение	Назначение
0	trustedToSecureRange0	При установке в 0 область доступна для secure обращений
1	trustedToSecureRange1	При установке в 0 область доступна для secure обращений
2	sdrToSecureRange0	При установке в 0 область доступна для secure обращений
3	sdrToSecureRange1	При установке в 0 область доступна для secure обращений
31:14	-	Резерв

3.9.2.6 Регистр TRUSTEDTOSECURE

Регистр изменения уровня доступа для областей памяти микросхемы с trusted на secure.

Формат регистра TRUSTEDTOSECURE приведен в таблице.

Таблица 3.24. Формат регистра TRUSTEDTOSECURE

Номер бита	Условное обозначение	Назначение
0	cpu_subs_sys_trusted	При установке в 0 область доступна для secure обращений
1	cpu_subs_debug_trusted	При установке в 0 область доступна для secure обращений
2	media_subs_sys_trusted	При установке в 0 область доступна для secure обращений
3	hsperiph_subs_sys_trusted	При установке в 0 область доступна для secure обращений
4	lsperiph0_subs_sys_trusted	При установке в 0 область доступна для secure обращений
5	lsperiph1_subs_sys_trusted	При установке в 0 область доступна для secure обращений
6	top_trusted	При установке в 0 область доступна для secure обращений
7	ddr_subs_ddr0_trusted	При установке в 0 область доступна для secure обращений
8	ddr_subs_ddr1_trusted	При установке в 0 область доступна для secure обращений
9	ddr_subs_sys_trusted	При установке в 0 область доступна для secure обращений
10	service_subs_mailbox0_sf_trusted	При установке в 0 область доступна для secure обращений
11	service_subs_sys_trusted	При установке в 0 область доступна для secure обращений
12	-	Резерв
13	sdr_subs_pcie_mid_trusted	При установке в 0 область доступна для secure обращений
31:14	-	Резерв

3.9.2.7 Регистр TRUSTED_ID_x

ID x для транзакций политики trusted.

По сбросу, регистры настроены на ID следующих мастеров:

TRUSTED_ID_0: RISC0/DEV
TRUSTED_ID_1: RISC0/DMA
TRUSTED_ID_2: MFBSP0
TRUSTED_ID_3: MFBSP1
TRUSTED_ID_4: Reserved
TRUSTED_ID_5: Reserved
TRUSTED_ID_6: Reserved
TRUSTED_ID_7: Reserved

Формат регистра TRUSTED_ID_x приведен в Таблица 3.25.

Таблица 3.25. Формат регистра TRUSTED_ID_x

Номер бита	Условное обозначение	Назначение
15:0	BASE	
31:16	MASK	

3.9.2.8 Регистр SDR_ID_x

ID x для транзакций политики sdr.

По сбросу, регистры настроены на ID следующих мастеров:

SDR_ID_0: DSP0/L2 + DSP1/L2
SDR_ID_1: DCC0/DMA + DCC1/DMA + DCC2/DMA + DCC3/DMA
SDR_ID_2: DUC/DMA + ADCBP/DMA
SDR_ID_3: DSP0/DMA + DSP1/DMA
SDR_ID_4: GNSS
SDR_ID_5: CMD/DMA + RISC1/DMA + RISC1/DEV
SDR_ID_6: MIX/DMA + RX/DMA
SDR_ID_7: AXI0/DMA + AXI1/DMA

Формат регистра SDR_ID_x приведен в Таблица 3.26.

Таблица 3.26. Формат регистра SDR_ID_x

Номер бита	Условное обозначение	Назначение
15:0	BASE	
31:16	MASK	

3.9.2.9 Регистр SLOW_AXI_DLOCK

Статус прерывания slow_axi_dlock.

Формат регистра SLOW_AXI_DLOCK приведен в Таблица 3.27.

Таблица 3.27. Формат регистра SLOW_AXI_DLOCK

Номер бита	Условное обозначение	Назначение
0	IRQ	Статус прерывания
1	WR	0: deadlock по чтению 1: deadlock по записи
3:2	-	Резерв
7:4	SLV	Номер slave порта коммутатора slow_axi: 0x0: Резерв 0x1: sdr_subs_sys 0x2: fast_comm 0x3: media_subs_sys 0x4: cpu_subs_sys 0x5: service_subs_sys 0x6: lsperiph0_subs_sys 0x7: lsperiph1_subs_sys 0x8: ddr_subs_sys 0x9: top_ctr 0xA-0xF: Резерв
11:8	MST	Номер мастер порта коммутатора slow_axi: 0x0: sdr_subs_sdr_dev 0x1: sdr_subs_pcie_mdev_slow 0x2: cpu_subs_dev_slow 0x3: service_subs_dev 0x4: lsperiph0_subs_dev 0x5: lsperiph1_subs_dev 0x6: hsperiph_subs_dev_slow 0x7-0xF: Резерв
15:12	-	Резерв
31:16	ID	ID deadlock транзакции.

3.9.2.10 Регистр FAST_AXI_DLOCK

Статус прерывания fast_axi_dlock.

Формат регистра FAST_AXI_DLOCK приведен в Таблица 3.28.

Таблица 3.28. Формат регистра FAST_AXI_DLOCK

Номер бита	Условное обозначение	Назначение
0	IRQ	Статус прерывания
1	WR	0: deadlock по чтению 1: deadlock по записи
3:2	-	Резерв

Номер бита	Условное обозначение	Назначение
7:4	SLV	Номер slave порта коммутатора fast_axi: 0x0: Резерв 0x1: sdr_subs_pcie_sdev 0x2: hsperiph_subs_sys 0x3-0xF: Резерв
11:8	MST	Номер мастер порта коммутатора fast_axi: 0x0: hsperiph_dev_fast 0x1: sdr_subs_pcie_mdev_fast 0x2: cpu_subs_dev_fast 0x3: slow_comm 0x4-0xF: Резерв
15:12	-	Резерв
31:16	ID	ID deadlock транзакции.

3.9.2.11 Регистр COH_AXI_DLOCK

Регистр управления когерентностью потоков данных.

Формат регистра COH_AXI_DLOCK приведен в Таблица 3.29.

Таблица 3.29. Формат регистра COH_AXI_DLOCK

Номер бита	Условное обозначение	Назначение
0	IRQ	Статус прерывания
1	WR	0: deadlock по чтению 1: deadlock по записи
3:2	-	Резерв
7:4	SLV	Номер slave-порта коммутатора coh_axi: 0x0: Резерв 0x1: cpu_subs_acp 0x2-0xF: Резерв
11:8	MST	Номер master-порта коммутатора coh_axi: 0x0: sdr_subs_sdr_ddr 0x1: sdr_subs_pcie_ddr 0x2: hsperiph_subs_ddr 0x3-0xF: Резерв
15:12	-	Резерв
31:16	ID	ID deadlock транзакции.

3.9.2.12 Регистр COH_REMAP

Регистр управления поддержкой когерентности.

Формат регистра COH_REMAP приведен в Таблица 3.30.

Таблица 3.30. Формат регистра COH_REMAP

Номер бита	Условное обозначение	Назначение
0	PCIE_REMAP	0: Когерентность канала выключена. Поток данных транслируется напрямую в ddr_subs. 1: Когерентность канала включена. Поток данных перенаправлен в порт acp cpu_subs.
1	SDR_REMAP	0: Когерентность канала выключена. Поток данных транслируется напрямую в ddr_subs. 1: Когерентность канала включена. Поток данных перенаправлен в порт acp cpu_subs.
2	HSPERIPH_REMAP	0: Когерентность канала выключена. Поток данных транслируется напрямую в ddr_subs. 1: Когерентность канала включена. Поток данных перенаправлен в порт acp cpu_subs.
31:3	-	Резерв

3.9.2.13 Регистр CHIP_VERSION

Регистр версии топа микросхемы.

Формат регистра CHIP_VERSION приведен в Таблица 3.31.

Таблица 3.31. Формат регистра CHIP_VERSION

Номер бита	Условное Обозначение	Назначение
15:0	TOP_ID	Версия chip_top определяется в виде r*x*p*y*, где *x*=TOP_ID [15:8], *y*=TOP_ID [7:0]
31:16	Резерв	

3.9.2.14 Регистр CPU_SUBS_VERSION

Регистр версии cpu_subs.

Формат регистра CPU_SUBS_VERSION приведен в Таблица 3.32.

Таблица 3.32. Формат регистра CPU_SUBS_VERSION

Номер бита	Условное Обозначение	Назначение
15:0	CPU_SUBS_ID	Версия cpu_subs определяется в виде r*x*p*y*, где *x*=CPU_SUBS_ID [15:8], *y*=CPU_SUBS_ID [7:0]
31:16	CPU_SUBS_FL_ID	Версия cpu_subs_fl определяется в виде r*x*p*y*, где *x*=CPU_SUBS_FL_ID [31:24], *y*=CPU_SUBS_FL_ID [23:16]

3.9.2.15 Регистр DDR_SUBS_VERSION

Регистр версии ddr_subs.

Формат регистра DDR_SUBS_VERSION приведен в Таблица 3.33.

Таблица 3.33. Формат регистра DDR_SUBS_VERSION

Номер бита	Условное Обозначение	Назначение
15:0	DDR_SUBS_ID	Версия cpu_subs определяется в виде r*x*p*y*, где *x*=DDR_SUBS_ID [15:8], *y*=DDR_SUBS_ID [7:0]
31:16	DDR_SUBS_FL_ID	Версия cpu_subs_fl определяется в виде r*x*p*y*, где *x*=DDR_SUBS_FL_ID [31:24], *y*=DDR_SUBS_FL_ID [23:16]

3.9.2.16 Регистр INTERCONNECT_SUBS_VERSION

Регистр версии interconnect_subs и библиотеки линков fl_lib.

Формат регистра INTERCONNECT_SUBS_VERSION приведен в Таблица 3.34.

Таблица 3.34. Формат регистра INTERCONNECT_SUBS_VERSION

Номер бита	Условное Обозначение	Назначение
15:0	INTERCONNECT_SUBS_ID	Версия interconnect_subs определяется в виде r*x*p*y*, где *x*=INTERCONNECT_SUBS_ID [15:8], *y*=INTERCONNECT_SUBS_ID [7:0]
31:16	FL_LIB_ID	Версия fl_lib определяется в виде r*x*p*y*, где *x*=FL_LIB_ID [31:24], *y*=FL_LIB_ID [23:16]

3.9.2.17 Регистр SDR_SUBS_VERSION

Регистр версии sdr_subs.

Формат регистра SDR_SUBS_VERSION приведен в Таблица 3.35.

Таблица 3.35. Формат регистра SDR_SUBS_VERSION

Номер бита	Условное Обозначение	Назначение
15:0	SDR_SUBS_ID	Версия sdr_subs определяется в виде r*x*p*y*, где *x*=SDR_SUBS_ID [15:8], *y*=SDR_SUBS_ID [7:0]
31:16	SDR_SUBS_FL_ID	Версия sdr_subs_fl определяется в виде r*x*p*y*, где *x*=SDR_SUBS_FL_ID [31:24], *y*=SDR_SUBS_FL_ID [23:16]

3.9.2.18 Регистр MEDIA_SUBS_VERSION

Регистр версии media_subs.

Формат регистра MEDIA_SUBS_VERSION приведен в Таблица 3.36.

Таблица 3.36. Формат регистра MEDIA_SUBS_VERSION

Номер бита	Условное Обозначение	Назначение
15:0	MEDIA_SUBS_ID	Версия media_subs определяется в виде r*x*p*y*, где *x*=MEDIA_SUBS_ID [15:8], *y*=MEDIA_SUBS_ID [7:0]
31:16	MEDIA_SUBS_FL_ID	Версия media_subs_fl определяется в виде r*x*p*y*, где *x*=MEDIA_SUBS_FL_ID [31:24], *y*=MEDIA_SUBS_FL_ID [23:16]

3.9.2.19 Регистр SERVICE_SUBS_VERSION

Регистр версии service_subs.

Формат регистра SERVICE_SUBS_VERSION приведен в Таблица 3.37.

Таблица 3.37. Формат регистра SERVICE_SUBS_VERSION

Номер бита	Условное Обозначение	Назначение
15:0	SERVICE_SUBS_ID	Версия service_subs определяется в виде r*x*p*y*, где *x*=SERVICE_SUBS_ID [15:8], *y*=SERVICE_SUBS_ID [7:0]
31:16	SERVICE_SUBS_FL_ID	Версия service_subs_fl определяется в виде r*x*p*y*, где *x*=SERVICE_SUBS_FL_ID [31:24], *y*=SERVICE_SUBS_FL_ID [23:16]

3.9.2.20 Регистр HSPERIPH_SUBS_VERSION

Регистр версии hsperiph_subs.

Формат регистра HSPERIPH_SUBS_VERSION приведен в Таблица 3.38.

Таблица 3.38. Формат регистра HSPERIPH_SUBS_VERSION

Номер бита	Условное Обозначение	Назначение
15:0	HSPERIPH_SUBS_ID	Версия hsperiph_subs определяется в виде r*x*p*y*, где *x*= HSPERIPH_SUBS_ID [15:8], *y*= HSPERIPH_SUBS_ID [7:0]
31:16	HSPERIPH_SUBS_FL_ID	Версия hsperiph_subs_fl определяется в виде r*x*p*y*, где *x*= HSPERIPH_SUBS_FL_ID [31:24], *y*= HSPERIPH_SUBS_FL_ID [23:16]

3.9.2.21 Регистр LSPERIPH0_SUBS_VERSION

Регистр версии lsperiph0_subs.

Формат регистра LSPERIPH0_SUBS_VERSION приведен в Таблица 3.39.

Таблица 3.39. Формат регистра LSPERIPH0_SUBS_VERSION

Номер бита	Условное Обозначение	Назначение
15:0	LSPERIPH0_SUBS_ID	Версия lsperiph0_subs определяется в виде r*x*p*y*, где *x*= LSPERIPH0_SUBS_ID [15:8], *y*= LSPERIPH0_SUBS_ID [7:0]
31:16	LSPERIPH0_SUBS_FL_ID	Версия lsperiph0_subs_fl определяется в виде r*x*p*y*, где *x*= LSPERIPH0_SUBS_FL_ID [31:24], *y*= LSPERIPH0_SUBS_FL_ID [23:16]

3.9.2.22 Регистр LSPERIPH1_SUBS_VERSION

Регистр версии lsperiph1_subs.

Формат регистра LSPERIPH1_SUBS_VERSION приведен в Таблица 3.40.

Таблица 3.40. Формат регистра LSPERIPH1_SUBS_VERSION

Номер бита	Условное Обозначение	Назначение
15:0	LSPERIPH1_SUBS_ID	Версия lsperiph1_subs определяется в виде r*x*p*y*, где *x*= LSPERIPH1_SUBS_ID [15:8], *y*= LSPERIPH1_SUBS_ID [7:0]
31:16	LSPERIPH1_SUBS_FL_ID	Версия lsperpih1_subs_fl определяется в виде r*x*p*y*, где *x*= LSPERIPH1_SUBS_FL_ID [31:24], *y*= LSPERIPH1_SUBS_FL_ID [23:16]

3.9.3 Диапазоны ID

Таблица 3.41. Принадлежность системных ID мастерам микросхемы Гиперком У.

Подсистема	Порт	Мастер	ID
cpu_subs	slow_dev, fast_dev	GIC500	13'b0_1000_00xx_xxxx
		CPU	13'b0_1000_01xx_xxxx
service_subs	Dev	RISC0 DEV	13'b0_1100_0000_000x
		RISC0 DMA	13'b0_1100_0000_001x
		MFBSP0	13'b0_1100_0000_010x
		MFBSP1	13'b0_1100_0000_011x
sdr_subs	pcie_dev_slow, pcie_dev_fast	PCI0	13'b0_0100_01xx_xxxx
		PCI1	13'b0_0100_10xx_xxxx
	sdr_dev	DSP0 L2 Cache	13'b0_0000_0000_xxxx
		DSP1 L2 Cache	13'b0_0001_0000_xxxx
		NELS DMA DDC0	13'b0_0010_0000_000x
		NELS DMA DDC1	13'b0_0010_0001_000x
		NELS DMA DDC2	13'b0_0010_0010_000x
		NELS DMA DCC3	13'b0_0010_0011_000x
		NELS DMA ADCBP	13'b0_0010_0100_000x
		NELS DMA DUC	13'b0_0010_0101_000x
		DSP0 DMA	13'b0_0010_0110_00xx
		DSP1 DMA	13'b0_0010_0111_00xx
		NAVICORE	13'b0_0010_1000_xxxx
		NELS DMA CMD	13'b0_0010_1001_000x
		RISC1 DMA	13'b0_0010_1001_001x
		RISC1 DEV	13'b0_0010_1001_010x
		ALT DMA RX	13'b0_0010_1010_000x

		ALT DMA MIX	13'b0_0010_1011_000x
		SDR MEM DMA0	13'b0_0010_1100_000x
		SDR MEM DMA1	13'b0_0010_1101_000x
hsperiph_subs	Dev	EMAC0	13'b0_0011_0000_xxxx
		EMAC1	13'b0_0011_0001_xxxx
		SDMMC0	13'b0_0011_0010_xxxx
		SDMMC1	13'b0_0011_0011_xxxx
		NFC	13'b0_0011_0100_xxxx
		USB0	13'b0_0011_0101_xxxx
		USB1	13'b0_0011_0110_xxxx
		PDMA2	13'b0_0011_0111_xxxx
		USOC	13'b0_0011_1000_xxxx
lsperiph0_subs	Dev	PDMA0	13'b1_0000_0000_000x
lsperiph1_subs	Dev	PDMA1	13'b1_0100_0000_000x

4. ПОДСИСТЕМА ЦЕНТРАЛЬНОГО ПРОЦЕССОРА

4.1 Общая информация

Подсистема центрального процессора (cpu_subs) – это часть микросхемы, которая включает в себя компоненты связанные с обеспечением работы центрального процессора (CPU).

Структурная схема подсистемы показана на Рисунок 4.1.

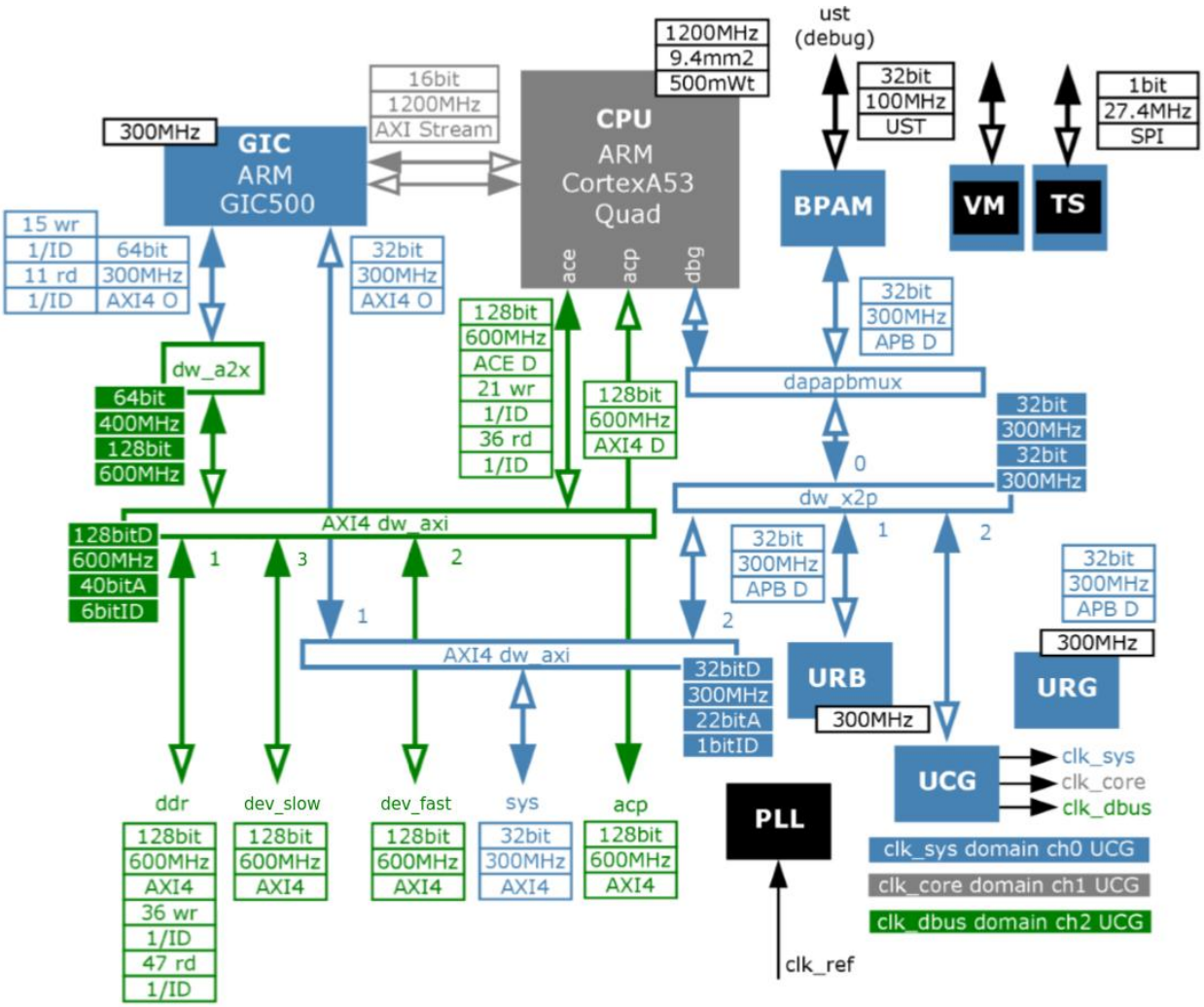


Рисунок 4.1. Структурная схема подсистемы

В состав подсистемы входят:

- CPU – центральный процессор, четырехъядерный кластер CortexA53;
- GIC – контроллер прерываний центрального процессора;
- BPAM (BPAM_CPU) – модуль отладки процессора, компонент системы отладки UltraSoC;

- PLL подсистемы;
- URB – блок управляющих регистров подсистемы;
- UCG – блок формирования тактовых частот подсистемы;
- URG – блок формирования сигналов сброса подсистемы;
- Локальный коммутатор в виде набора компонент;

4.2 Система синхронизации

Система синхронизации подсистемы обеспечивается блоками PLL и UCG. На основе базовой частоты clk_ref с уровня SoC с помощью PLL обеспечивается формирование опорной частоты для UCG, соответствующей наиболее высокой частоте в подсистеме – частоте ядра CPU. Далее UCG формирует внутренние частоты подсистемы. Схема формирования частот показана на Рисунок 4.2.

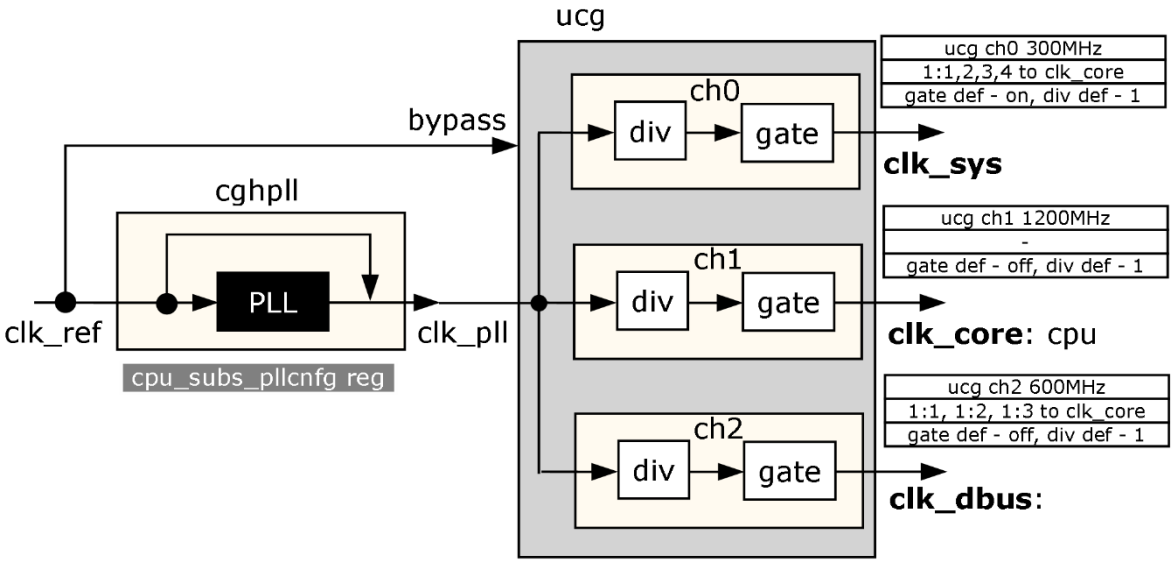


Рисунок 4.2. Схема формирования частот подсистемы

Подсистема имеет следующие внешние источники частоты:

- clk_ref – опорная частота подсистемы с уровня SoC, соответствует опорной частоте микросхемы – 27.456 МГц.
- uct_clk – частота внешней интерфейсной части BPAM, соединенной с коммутатором отладочной логики уровня микросхемы.

PLL подсистемы позволяет генерировать частоты в диапазоне от опорной частоты микросхемы до частоты 3.185 ГГц.

В Таблица 4.1 дано описание тактовых частот подсистемы.

Таблица 4.1. Описание тактовых частот микросхемы

Название	Описание	Максимальные рабочие значения при НУ/WC	Ограничения	Источник
Внешние источники частоты				
clk_ref	Опорная частота PLL подсистемы. Общая частота подсистемы после сброса.	27.456 МГц		SoC
ust_clk	Частота работы интерфейса с SoC блока BPAM, частота работы отладочной логики UltraSoC микросхемы.	100 МГц		SoC
Внутренние частоты				
clk_core	Основная частота центрального процессора	2032 / 1120 МГц *		UCG ch1
clk_dbus	Частота портов dev, ddr, acp подсистемы. Частота коммутационной логики, связанной с этими портами.	1016 / 590 МГц	Может относиться к clk_core как 1:2	UCG ch2
clk_sys	Частота работы системной части подсистемы – блоков PLL, URB, URG, UCG, порта подсистемы sys и логики коммутатора, связанного с этим портом.	678 / 295 МГц	Может относиться к clk_core как 1:4	UCG ch0

* Частота ARM® Cortex®-A53 MPCore - не менее 1120 МГц. Подлежит уточнению после получения и измерения макетных образцов микросхемы К1892ВМ21Я.

В состоянии после сброса каналы формирования частот 1 и 2 UCG выключены и частоты не сконфигурированы, системная часть подсистемы, т.е. домен clk_sys и канал 0 включен работает на опорной частоте clk_ref в режиме bypass. Ограничения на соотношение частот перечисленные в Таблица 4.1 должны соблюдаться при настройке UCG. Несоблюдение может привести к неработоспособности микросхемы. Обязательно выполнение процедуры синхронизации каналов (п. 43.3.7 Синхронизация каналов).

4.3 Логика сброса

4.3.1 Внешние источники сброса

Подсистема работает по следующим источникам сброса с уровня СнК:

- rstn_ref – общий сигнал сброса подсистемы с уровня СнК, ассинхнонный, активный уровень – 0, синхронное снятие по частоте clk_ref;
- ust_rst - внешней интерфейсной части BPAM, соединенной с коммутатором отладочной логики уровня микросхемы, ассинхнонный, активный уровень – 1, синхронное снятие по частоте ust_clk.

4.3.2 Внутренняя генерация сброса

Различные компоненты подсистемы могут сбрасываться по следующим причинам:

- Внешние источники сброса;
- Запросы на warm reset со стороны ядер CPU;
- Регистры PPU URB подсистемы (см. регистры a53cpuN_ppolicy URB);

В Таблица 4.2 показаны внутренние источники сброса и причины по которым происходит их сброс. Таблица использует следующую нотацию:

- Да – сброс является прямым следствием соблюдения условия;
- Нет – условие не приводит к сбросу;
- P – условие приводит к сбросу не напрямую, а через сброс регистров PP, который в свою очередь инициирует процедуру сброса.

Таблица 4.2. Источники сброса в подсистеме

Компонент или сигнал сброса компонента	Внутренние источники											Внешние источники	
	CPU0 PP OFF	CPU0 PP WARM_ RST	CPU1 PP OFF	CPU1 PP WARM_ RST	CPU2 PP OFF	CPU2 PP WARM_ RST	CPU3 PP OFF	CPU3 PP WARM_ RST	CPU SYS PP OFF	CPU SYS PP WARM_ RST	Другие регистры	rstn_ref	ust_rst
CPU.nCOREPORRESET[0]	Да	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	P	Нет
CPU.nCORERESET[0]	Да	Да	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	EDPRCR, RMR - P	P	Нет
CPU.nCOREPORRESET[1]	Нет	Нет	Да	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	P	Нет
CPU.nCORERESET[1]	Нет	Нет	Да	Да	Нет	Нет	Нет	Нет	Нет	Нет	EDPRCR, RMR - P	P	Нет
CPU.nCOREPORRESET[2]	Нет	Нет	Нет	Нет	Да	Нет	Нет	Нет	Нет	Нет	Нет	P	Нет
CPU.nCORERESET[2]	Нет	Нет	Нет	Нет	Да	Да	Нет	Нет	Нет	Нет	EDPRCR, RMR - P	P	Нет
CPU.nCOREPORRESET[3]	Нет	Нет	Нет	Нет	Нет	Нет	Да	Нет	Нет	Нет	Нет	P	Нет
CPU.nCORERESET[3]	Нет	Нет	Нет	Нет	Нет	Нет	Да	Да	Нет	Нет	EDPRCR, RMR - P	P	Нет
CPU.nL2RESET	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Да	Да	Нет	P	Нет
CPU.nPRESETDBG	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Да	Нет	Нет	P	Нет
GIC	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Да	Нет
BPAM	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	debug_reset message	Да	Да
URB	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Да	Нет
UCG	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Да	Нет
Interconnect	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Да	Нет

4.3.3 Старт ядер CPU после сброса микросхемы

После сброса микросхемы и подсистемы ядра кластера (CPU0-3) и его системная часть находятся в состоянии определяемым начальными значениями регистров a53cpuN_ppolicy и a53sys_ppolicy. В данной конфигурации подсистемы начальные значения этих регистров установлены в резервном значении - 0, что означает состояние выключено для ядер и системной части кластера.

Рекомендуется следующий порядок старта для кластера CPU после сброса микросхемы:

- Настройка частот подсистемы;
    - Включение PLL на требуемую частоту работы PLL;
    - Настройка и включение через UCG частот clk_core, clk_sys, clk_dbus;
    - Переключение частоты clk_sys с источника от clk_ref на источник от UCG;
- Настройка (если требуется) конфигурационных параметров работы CPU, доступных через регистры URB;
- Включение системного домена CPU через регистр a53sys_ppolicy;
- Включение доменов ядер через регистры a53cpuN_ppolicy;

После включения ядро CPU начинает исполнение программы с заданного стартового адреса (по умолчанию – 0x00_0000_0000).

4.4 Интерфейсы подсистемы

Параметры интерфейсов подсистемы приведены в Таблица 4.3.

Таблица 4.3. Параметры интерфейсов подсистемы

Название интерфейса	Назначение	Тип	Разрядность шины данных	Разрядность ID	Число реально используемых ID	Максимальное число удаленных транзакций	Требования к латентности	Требования к пиковой пропускной способности
ddr	Доступ к ddr	AXI4 Master	128	7	36 wr 47 rd	15wr/4rd на ID 0b’011nn, 1 на остальные ID: ~64wr/64rd	Минимально возможная	50 Гбит/с
dev	Доступ к внутренней памяти и PCIe	AXI4 Master	128	7	36 wr 47 rd	15 на ID 0b’011nn, 1 на остальные ID: ~64wr/64rd	Минимально возможная	10 Гбит/с

Название интерфейса	Назначение	Тип	Разрядность шины данных	Разрядность ID	Число реально используемых ID	Максимальное число удаленных транзакций	Требования к латентности	Требования к пиковой пропускной способности
acp	Когерентный порт CPU	AXI4 Slave	128	5	-	4, 1 на ID	-	-
sys	Доступ к регистрам подсистемы	AXI4 Slave	32	TBD	-	-	-	-
ust	Отладка через UltraSoC	UST	32	-	-	-	-	-

4.5 Программная модель и регистры подсистемы

В Таблица 4.4 дана карта внутренней памяти подсистемы, доступная через ее порт sys.

Таблица 4.4. Карта внутренней памяти подсистемы

Стартовый адрес	Размер области	Описание
0x0000800000	1 Мбайт	Регистры интерфейса GIC CPU
0x0000900000	3 Мбайта	Резерв
0x0000c00000	4 Мбайта	Отладочная область и отладочные регистры CPU
0x0001000000	64 Кбайта	Регистры URB
0x0001010000	-	Резерв
0x0001080000	4 Кбайта	Регистры UCG
0x0001081000	-	Резерв
0x0001100000	1 Мбайт	Регистры GIC

Область отладочной логики DEBUG APB CPU доступна также по интерфейсу ust через компонент отладки BPAM.

Далее дано описание регистров URB, остальные регистры внутренней памяти описаны в главах, посвященных соответствующим блокам.

4.5.1 Регистры URB подсистемы

Перечень регистров Блок регистров URB cpu_subs приведен в следующей таблице:

Таблица 4.5. Перечень программно-доступных регистров Блок регистров URB cpu_subs

Смещение	Условное Обозначение	Название регистра	Исходное состояние	Тип доступа
0	a53cpu0_ppolicy	Регистр установки состояния ядра 0 CPU	0x0	RW
0x0004	a53cpu0_pstatus	Регистр статуса состояния ядра 0 CPU	0x0	RO
0x0010	a53cpu1_ppolicy	Регистр установки состояния ядра 1 CPU	0x0	RW
0x0014	a53cpu1_pstatus	Регистр статуса состояния ядра 1 CPU	0x0	RO
0x0020	a53cpu2_ppolicy	Регистр установки состояния ядра 2 CPU	0x0	RW
0x0024	a53cpu2_pstatus	Регистр статуса состояния ядра 2 CPU	0x0	RO
0x0030	a53cpu3_ppolicy	Регистр установки состояния ядра 3 CPU	0x0	RW
0x0034	a53cpu3_pstatus	Регистр статуса состояния ядра 3 CPU	0x0	RO
0x0040	a53sys_ppolicy	Регистр установки состояния кластерной части CPU	0x0	RW
0x0044	a53sys_pstatus	Регистр статуса состояния кластерной части CPU	0x0	RO
0x0050	cpu_subs_pllcnfg	Регистр управления PLL подсистемы	0x0	RW
0x0054	cpu_subs_plldiag	Регистр диагностики PLL подсистемы	0x0	RW
0x0060	a53cpu_pcnfg	Регистр конфигурации процедур включения и выключения доменов cpu	0x0	RW
0x0064	a53sys_pcnfg	Регистр конфигурации процедур включения и выключения домена sys	0x0	RW
0x0100	a53_rstcnfg	Регистр конфигурации сброса CPU	0x0	RW
0x0104	a53_acpcfg	Регистр конфигурации ACP порта CPU	0x0	RW
0x0108	gic500_status	Регистр статуса GIC500	0x0	RO
0x010c	cpu_eventimask	Регист маски события для CPU	0x0	RW
0x0110	cpu_l2cnfg	Регистр конфигурации L2 кэш CPU	0x0	RW
0x0114	cpu_startcfg	Регистр конфигурации старта CPU	0x0	RW
0x0118	cpu_rvbaddrh0	Регистр вектора сброса CPU0 старшая часть	0x0	RW
0x011c	cpu_rvbaddrl0	Регистр вектора сброса CPU0 младшая часть	0x0	RW
0x0120	cpu_rvbaddrh1	Регистр вектора сброса CPU1 старшая часть	0x0	RW
0x0124	cpu_rvbaddrl1	Регистр вектора сброса CPU1 младшая часть	0x0	RW
0x0128	cpu_rvbaddrh2	Регистр вектора сброса CPU2 старшая часть	0x0	RW
0x012c	cpu_rvbaddrl2	Регистр вектора сброса CPU2 младшая часть	0x0	RW

0x0130	cpu_rvbaddrh3	Регистр вектора сброса CPU3 старшая часть	0x0	RW
0x0134	cpu_rvbaddrl3	Регистр вектора сброса CPU3 младшая часть	0x0	RW

4.5.2 Описание регистра a53cpu0_ppolicy

Формат регистра a53cpu0_ppolicy приведен в следующей таблице:

Таблица 4.6. Поля регистра a53cpu0_ppolicy

Диапазон	Название	Описание	Сброс
31:5	-	Резерв	0
4:0	Policy	Установка состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; 'b00001 PP_OFF - выключено; другое - резерв/выключено	'b01000

4.5.3 Описание регистра a53cpu0_pstatus

Формат регистра a53cpu0_pstatus приведен в следующей таблице:

Таблица 4.7. Поля регистра a53cpu0_pstatus

Диапазон	Название	Описание	Сброс
31:9	-	Резерв	0
8	memretspt	Состояние MEM_RET поддерживается в установках для этого домена	0
7:6	-	Резерв	0
5	Emulated	Значение 1 указывает что состояние OFF эмулируется для нужд отладки	0
4:0	Pstatus	Поле отражает текущую активную установку состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; 'b00001 PP_OFF - выключено; другое - резерв/выключено	-

4.5.4 Описание регистра a53cpu1_ppolicy

Формат регистра a53cpu1_ppolicy приведен в следующей таблице:

Таблица 4.8. Поля регистра a53cpu1_ppolicy

Диапазон	Название	Описание	Сброс
31:5	-	Резерв	0
4:0	policy	Установка состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; 'b00001 PP_OFF - выключено; другое - резерв/выключено	'b01000

4.5.5 Описание регистра a53cpu1_pstatus

Формат регистра a53cpu1_pstatus приведен в следующей таблице:

Таблица 4.9. Поля регистра a53cpu1_pstatus

Диапазон	Название	Описание	Сброс
31:9	-	Резерв	0
8	memretspt	Состояние MEM_RET поддерживается в установках для этого домена	0
7:6	-	Резерв	0
5	emulated	Значение 1 указывает что состояние OFF эмулируется для нужд отладки	0
4:0	pstatus	Поле отражает текущую активную установку состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; 'b00001 PP_OFF - выключено; другое - резерв/выключено	-

4.5.6 Описание регистра a53cpu2_ppolicy

Формат регистра a53cpu2_ppolicy приведен в следующей таблице:

Таблица 4.10. Поля регистра a53cpu2_ppolicy

Диапазон	Название	Описание	Сброс
31:5	-	Резерв	0
4:0	policy	Установка состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; 'b00001 PP_OFF - выключено; другое - резерв/выключено	'b01000

4.5.7 Описание регистра a53cpu2_pstatus

Формат регистра a53cpu2_pstatus приведен в следующей таблице:

Таблица 4.11. Поля регистра a53cpu2_pstatus

Диапазон	Название	Описание	Сброс
31:9	-	Резерв	0
8	memretspt	Состояние MEM_RET поддерживается в установках для этого домена	0
7:6	-	Резерв	0
5	emulated	Значение 1 указывает что состояние OFF эмулируется для нужд отладки	0
4:0	pstatus	Поле отражает текущую активную установку состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; 'b00001 PP_OFF - выключено; другое - резерв/выключено	-

4.5.8 Описание регистра a53cpu3_ppolicy

Формат регистра a53cpu3_ppolicy приведен в следующей таблице:

Таблица 4.12. Поля регистра a53cpu3_ppolicy

Диапазон	Название	Описание	Сброс
31:5	-	Резерв	0
4:0	policy	Установка состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; 'b00001 PP_OFF - выключено; другое - резерв/выключено	'b01000

4.5.9 Описание регистра a53cpu3_pstatus

Формат регистра a53cpu3_pstatus приведен в следующей таблице:

Таблица 4.13. Поля регистра a53cpu3_pstatus

Диапазон	Название	Описание	Сброс
31:9	-	Резерв	0
8	memretspt	Состояние MEM_RET поддерживается в установках для этого домена	0
7:6	-	Резерв	0
5	emulated	Значение 1 указывает что состояние OFF эмулируется для нужд отладки	0
4:0	pstatus	Поле отражает текущую активную установку состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; 'b00001 PP_OFF - выключено; другое - резерв/выключено	-

4.5.10 Описание регистра a53sys_ppolicy

Формат регистра a53sys_ppolicy приведен в следующей таблице:

Таблица 4.14. Поля регистра a53sys_ppolicy

Диапазон	Название	Описание	Сброс
31:5	-	Резерв	0
4:0	policy	Установка состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; 'b00001 PP_OFF - выключено; другое - резерв/выключено	'b01000

4.5.11 Описание регистра a53sys_pstatus

Формат регистра a53sys_pstatus приведен в следующей таблице:

Таблица 4.15. Поля регистра a53sys_pstatus

Диапазон	Название	Описание	Сброс
31:9	-	Резерв	0
8	memretspt	Состояние MEM_RET поддерживается в установках для этого домена	0
7:6	-	Резерв	0
5	emulated	Значение 1 указывает что состояние OFF эмулируется для нужд отладки	0
4:0	pstatus	Поле отражает текущую активную установку состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; 'b00001 PP_OFF - выключено; другое - резерв/выключено	-

4.5.12 Описание регистра a53cpu_pcnfg

Формат регистра a53cpu_pcnfg приведен в следующей таблице:

Таблица 4.16. Поля регистра a53cpu_pcnfg

Диапазон	Название	Описание	Сброс
31:20	-	Резерв	?
19:16	delay_disable	отключение задержки между включением отдельных цепей питания домена	'b1101
15:0	delay	задержка между включением цепей питания домена	0xffff

4.5.13 Описание регистра a53sys_pcnfg

Формат регистра a53sys_pcnfg приведен в следующей таблице:

Таблица 4.17. Поля регистра a53sys_pcnfg

Диапазон	Название	Описание	Сброс
31:20	-	Резерв	?
19:16	delay_disable	отключение задержки между включением отдельных цепей питания домена	'b1101
15:0	delay	задержка между включением цепей питания домена	0xffff

4.5.14 Описание регистра cpu_subs_pllcnfg

Формат регистра cpu_subs_pllcnfg приведен в следующей таблице:

Таблица 4.18. Поля регистра cpu_subs_pllcnfg

Диапазон	Название	Описание	Сброс
31	lock	Бит указывает окончание настройки PLL на новую заданную частоту работы. При начале процедуры настройки либо при выключенной PLL сбрасывается в 0.	0
30:27	nr_man	Коэффициент nr настройки pll	
26:14	nf_man	Коэффициент nf настройки pll	0
13:10	od_man	Коэффициент od настройки pll	0

Диапазон	Название	Описание	Сброс
9:9	man	При бите заданном в 1 и sel > 0 конфигурация pll определяется полям *_man данного регистра. Выходная частота определяется по формуле ref_clk * (nf_man + 1) / (nr_man + 1) / (od_man + 1). Поля *_man рекомендуется задавать исходя из значений генерируемым скриптом настройки pll. Значение выходной частоты не должно превышать 3600 МГц.	0
8:8	-	Резерв	0
7:0	sel	Определяет выходную частоту и режим работы PLL: 0x0 - опорная частота ref_clk, PLL выключена; 0x1 - ref_clk * 2; 0x2 - ref_clk * 3; 0x3 - ref_clk * 4; ... - ...; 0x73 - ref_clk * 116; 0x74 - ref_clk * 116; ... - ...; 0xFF - ref_clk * 116; При sel > 0 и man = 1, конфигурация pll определяется полями *_man данного регистра	0

4.5.15 Описание регистра cpu_subs_plldiag

Формат регистра cpu_subs_plldiag приведен в следующей таблице:

Таблица 4.19. Поля регистра cpu_subs_plldiag

Диапазон	Название	Описание	Сброс
31:5	-	Резерв	
4:4	fbslip	Диагностический выход pll - регистрирует ситуацию feedback cycle slip	0
3:3	rfslip	Диагностический выход pll - регистрирует ситуацию referency cycle slip	0
2:2	fasten	Диагностический вход pll - разрешает механизм fast lock	0
1:1	ensat	Диагностический вход pll - разрешает насыщение, в нормальном режиме - 1	1
0:0	test	Диагностический вход pll - bypass опорной частоты pll на выход при 1	0

4.5.16 Описание регистра a53_rstcnfg

Формат регистра a53_rstcnfg приведен в следующей таблице:

Таблица 4.20. Поля регистра a53_rstcnfg

Диапазон	Название	Описание	Сброс
31:16	-	Резерв	0
15:12	dbgrstreqen	Включает возможность сброса ядра через регистр EDPRCR в CP14. Соответсвие битов поля ядрам: [15:12] - [3:0].	0
11:8	warmrstreqen	Включает возможность сброса ядра через регистр RMR CPU. Соответсвие битов поля ядрам: [11:8] - [3:0].	0
7:2	-	Резерв	0

Диапазон	Название	Описание	Сброс
1	l2rstdisable	Отключает автоматическую инвалидацию кэша L2 по сбросу. Запись в этот регистр допускается только когда CPU в состоянии сброса	0
0	dbgl1rstdisable	Отключает автоматическую инвалидацию кэша данных L1 по сбросу. Значение регисрируется только во время сброса CPU. Номер бита соответсвует номеру ядра.	0

4.5.17 Описание регистра a53_acpcfg

Формат регистра a53_acpcfg приведен в следующей таблице:

Таблица 4.21. Поля регистра a53_acpcfg

Диапазон	Название	Описание	Сброс
31:5	-	Резерв	0
4:3	awuser	Поле задает для CPU значение входа AWUSER ACP порта. Вход задает атрибуты shareability для всех запросов по записи в порт: 0b00 - non-sharable; 0b01 - inner sharable; 0b10 - outer sharable	0
2:1	aruser	Поле задает для CPU значение входа ARUSER ACP порта. Вход задает атрибуты shareability для всех запросов по чтению в порт: 0b00 - non-sharable; 0b01 - inner sharable; 0b10 - outer sharable;	0
0	ainact	Бит отражает для CPU включение функции IO когерентности в микросхеме через ACP порт: 0 - функция включена; 1 - функция отключена; Значение бита должно соответствовать настройкам когерентности микросхемы на уровне системы коммутации.	1

4.5.18 Описание регистра gic500_status

Формат регистра gic500_status приведен в следующей таблице:

Таблица 4.22. Поля регистра gic500_status

Диапазон	Название	Описание	Сброс
31:2	-	Резерв	0
1	gic_axim_err	Ошибка доступа по шине при обращении со стороны GIC500. Требуется сброс подсистемы.	0
0	gic_ecc_fatal	Ошибка ECC в работе внутренней памяти GIC500. Требуется сброс подсистемы.	0

4.5.19 Описание регистра cpu_eventimask

Формат регистра cpu_eventimask приведен в следующей таблице:

Таблица 4.23. Поля регистра cpu_eventimask

Диапазон	Название	Описание	Сброс
31:4	-	Резерв	0
3	dsp1eventen	маска события от DSP1	0
2	dsp0eventen	маска события от DSP0	0
1	risc1eventen	маска события от RISC1	0
0	risc0eventen	маска события от RISC0	0

4.5.20 Описание регистра cpu_l2cnfg

Формат регистра cpu_l2cnfg приведен в следующей таблице:

Таблица 4.24. Поля регистра cpu_l2cnfg

Диапазон	Название	Описание	Сброс
31:1	-	Резерв	0
0	l2flush	Запись 1 в этот бит инициирует процедуру flush кэша L2 CPU. Бит читается единицей до окончания процедуры и нулем после ее окончания.	0

4.5.21 Описание регистра cpu_startcfg

Формат регистра cpu_startcfg приведен в следующей таблице:

Таблица 4.25. Поля регистра cpu_startcfg

Диапазон	Название	Описание	Сброс
31:20	-	Резерв	0
19:16	vinithi	Задает базовый адрес векторов исключений по сбросу. Задает начальное значение бита V в регистрах SCTLR и HSCTLR. 0 - вектора исключений с адреса 0x00000000; 1 - вектора исключений с адреса 0xFFFF0000. Значение регистрируется в CPU только во время сброса. 4 бита поля задают значения для 4 ядер кластера.	0
15:12	cp15disable	Запрещает доступ по записи в некоторы secure регистры CP15. Значение регистрируется в CPU только во время сброса. 4 бита поля задают значения для 4 ядер кластера.	0
11:8	cfgte	Разрешает исключения T32. Задает начальное значение поля бита TE в регистрах SCTLR и HSCTLR. 0 - бит TE в нуле; 1 - бит TE в единице. Значение регистрируется в CPU только во время сброса. 4 бита поля задают значения для 4 ядер кластера.	0
7:4	cfgend	Задает порядок байт в слове для данных во время обработки исключений. Задает значение битов EE в регистрах SCTLR, HSCTLR, SCTLR_EL1, SCTLR_EL2, SCTLR_EL3. 0 - бит EE в нуле (LittleEndian); 1 - бит EE в единице (BigEndian). Значение регистрируется в CPU только во время сброса. 4 бита поля задают значения для 4 ядер кластера.	0

Диапазон	Название	Описание	Сброс
3:0	aa64naa32	Разрешает состояние 64-бита для регистров CPU, при сбросе по включению питания. 0 - AArch32, 1 - AArch64. Значение регистрируется в CPU только во время сброса. 4 бита поля задают значения для 4 ядер кластера.	0

4.5.22 Описание регистра cpu_rvbaddrh0

Формат регистра cpu_rvbaddrh0 приведен в следующей таблице:

Таблица 4.26. Поля регистра cpu_rvbaddrh0

Диапазон	Название	Описание	Сброс
31:8	-	Резерв	0
7:0	rvbaddr39_32	Задает биты [39:32] адреса вектора сброса CPU в 64-битном режиме	0

4.5.23 Описание регистра cpu_rvbaddrl0

Формат регистра cpu_rvbaddrl0 приведен в следующей таблице:

Таблица 4.27. Поля регистра cpu_rvbaddrl0

Диапазон	Название	Описание	Сброс
31:2	rvbaddr31_2	Задает биты [31:2] адреса вектора сброса CPU в 64-битном режиме	0
1:0	-	Резерв	0

4.5.24 Описание регистра cpu_rvbaddrh1

Формат регистра cpu_rvbaddrh1 приведен в следующей таблице:

Таблица 4.28. Поля регистра cpu_rvbaddrh1

Диапазон	Название	Описание	Сброс
31:8	-	Резерв	0
7:0	rvbaddr39_32	Задает биты [39:32] адреса вектора сброса CPU в 64-битном режиме	0

4.5.25 Описание регистра cpu_rvbaddrl1

Формат регистра cpu_rvbaddrl1 приведен в следующей таблице:

Таблица 4.29. Поля регистра cpu_rvbaddrl1

Диапазон	Название	Описание	Сброс
31:2	rvbaddr31_2	Задает биты [31:2] адреса вектора сброса CPU в 64-битном режиме	0
1:0	-	Резерв	0

4.5.26 Описание регистра cpu_rvbaddrh2

Формат регистра cpu_rvbaddrh2 приведен в следующей таблице:

Таблица 4.30. Поля регистра cpu_rvbaddrh2

Диапазон	Название	Описание	Сброс
31:8	-	Резерв	0
7:0	rvbaddr39_32	Задает биты [39:32] адреса вектора сброса CPU в 64-битном режиме	0

4.5.27 Описание регистра cpu_rvbaddrl2

Формат регистра cpu_rvbaddrl2 приведен в следующей таблице:

Таблица 4.31. Поля регистра cpu_rvbaddrl2

Диапазон	Название	Описание	Сброс
31:2	rvbaddr31_2	Задает биты [31:2] адреса вектора сброса CPU в 64-битном режиме	0
1:0	-	Резерв	0

4.5.28 Описание регистра cpu_rvbaddrh3

Формат регистра cpu_rvbaddrh3 приведен в следующей таблице:

Таблица 4.32. Поля регистра cpu_rvbaddrh3

Диапазон	Название	Описание	Сброс
31:8	-	Резерв	0
7:0	rvbaddr39_32	Задает биты [39:32] адреса вектора сброса CPU в 64-битном режиме	0

4.5.29 Описание регистра cpu_rvbaddrl3

Формат регистра cpu_rvbaddrl3 приведен в следующей таблице:

Таблица 4.33. Поля регистра cpu_rvbaddrl3

Диапазон	Название	Описание	Сброс
31:2	rvbaddr31_2	Задает биты [31:2] адреса вектора сброса CPU в 64-битном режиме	0
1:0	-	Резерв	0

4.6 Дополнительная информация

4.6.1 Трансляция DeviceID PCIe для ITS записей в GIC500

Прерывания в GIC500 могут генерироваться с помощью записей в ITS GIC500, для того чтобы обеспечить различные правила трансляции прерываний от различных устройств в GIC500 вместе с записью должен передаваться ID устройства. В микросхеме реализована поддержка этого механизма, в частности, для блока PCIe0.

Для этого необходимо провести трансляцию выхода контроллера PCIe0 mstr_awmisc_info[39:24] с requester ID на вход awuser[19:4] GIC500 (согласно GIC Integration Manual стр. 2-5, 2-6). Такая трансляция необходима только при записи в регистр GITS_TRANSLATOR GIC по адресу 0x00_0113_0040, притом что во всем диапазоне 0x00_0113_0000 – 0x00_0113_ffff доступен только этот один регистр. Поэтому в обертке контроллера PCIe при записи по адресам со старшей частью равной 0x00_0013 производится подстановка поля mstr_awmisc_info[39:24] в младшую часть адреса. Со стороны GIC при записи с awid соответствующей PCIe0 и со старшей частью адреса равной 0x00_0013 производится подстановка awaddr[15:0] на вход awuser[19:4], awaddr[15:0] в свою очередь заменяется на 0x0040. В случае если awid записи не равен ID PCIe контроллеров, awuser[20] присваивается 1, awuser[19:15] – 0, awuser[14:4] – awid записи.

5. СЛУЖЕБНАЯ ПОДСИСТЕМА

5.1 Общая информация

Служебная подсистема (service_subs) – это часть микросхемы, которая реализует доверенный контур микросхемы (см. Раздел 3.2) в том числе следующие его основные функции:

- Управление общими настройками микросхемы;
- Управление и генерация опорных тактовых частот для остальных подсистем микросхемы;
- Управление доменами питания микросхемы, включением и выключением остальных подсистем;
- Начальная загрузка и настройка микросхемы;
- Управление отладкой микросхемы;
- Предоставляет аппаратные компоненты необходимые для исполнения доверенной среды исполнения и ее сервисов.

Структурная схема подсистема показана на Рисунок 5.1.

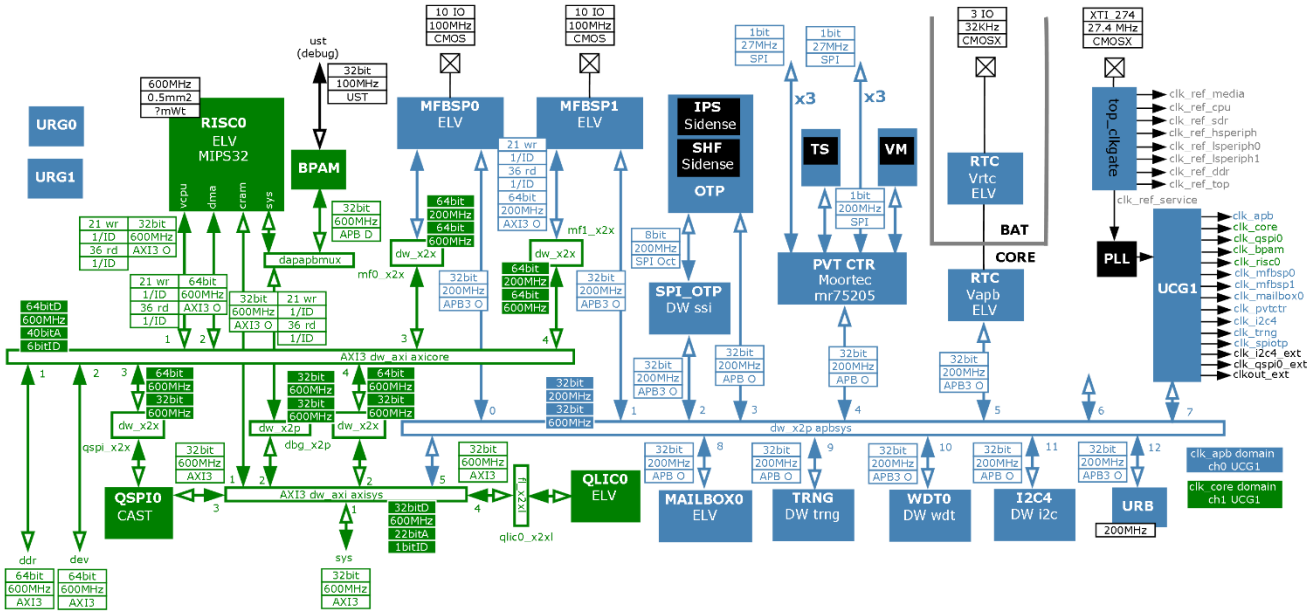


Рисунок 5.1. Структурная схема подсистемы

В состав подсистемы входят:

- RISC0 – доверенное процессорное ядро;

- QLIC0 – контроллер доверенных прерываний;
- OTP – доверенное хранилище ключевой информации;
- RTC – доверенный таймер реального времени;
- WDT0 – доверенный watchdog таймер;
- TRNG – доверенный генератор случайных чисел;
- PVT CTR – контроллер накристальных сенсоров температуры и напряжения микросхемы;
- MFBSP0 – доверенный контроллер периферии или SPI Flash памяти;
- MFBSP1 – доверенный контроллер периферии или SPI Flash памяти;
- QSPI0 – доверенный контроллер SPI Flash памяти с функцией непосредственного исполнения;
- BPAM (BPAM_SERV) – модуль отладки процессора RISC0, компонент системы отладки UltraSoC;
- PLL подсистемы;
- URB – блок управляющих регистров подсистемы и микросхемы;
- top_clkgate – набор аппаратных выключателей для формирования опорных тактовых частот для подсистем микросхемы из XTI_274;
- UCG1 – блок формирования тактовых частот подсистемы;
- URG0 – блок формирования глобальных сигналов сброса для подсистем микросхемы;
- URG1 – блок формирования сигналов сброса подсистемы;
- SPI_OTP – внутренний контроллер SPI для обращений к OTP памяти;
- MAILBOX0 – почтовый ящик для обращений к доверенному контуру;
- I2C4 – контроллер I2C для связи с микросхемами управления электропитанием (PMIC);
- Локальный коммутатор в виде набора компонент;

5.2 Система синхронизации

5.2.1 Формирование опорных тактовых частот для подсистем микросхемы

Опорные тактовые частоты всех подсистем микросхемы формируются набором выключателей, в рамках служебной подсистемы, на основе внешнего источника с выводов XTI274/XTO274 микросхемы (см. Рисунок 5.1).

Формирование тактовых частот для не предполагает умножения частоты, только их выключение, поэтому опорные тактовые частоты подсистем равны частоте внешнего источника (по умолчанию – 27.456 МГц).

Выключение и включение опорных тактовых частот подсистем производится через регистр top_clkgate. Механизма безопасного включения отключения частот не предусмотрено, поэтому стоит убедиться в отсутствии активных транзакций для отключаемой подсистемы.

5.2.2 Формирование опорных тактовых частот для блоков служебной подсистемы

Система синхронизации подсистемы обеспечивается блоками PLL и UCG1. На основе базовой частоты clk_ref_service с помощью PLL обеспечивается формирование опорной частоты для UCG1, соответствующей наиболее высокой частоте в подсистеме – частоте ядра RISC0. Далее UCG1 формирует внутренние частоты подсистемы. Схема формирования частот показана на Рисунок 5.2.

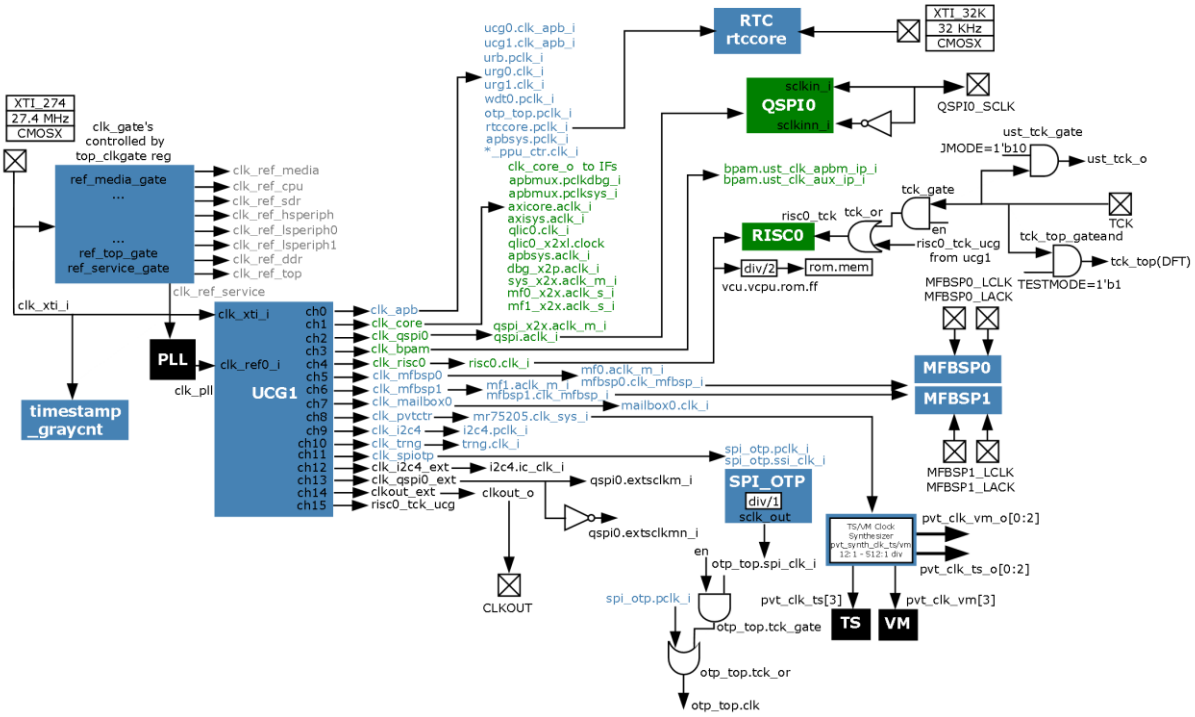


Рисунок 5.2. Схема формирования частот в подсистеме

Подсистема имеет следующие внешние источники частоты:

- clk_ref_service – опорная частота подсистемы, формируемая через clk_gate от опорной частоты микросхемы XTI_274 – 27.456 МГц.
- ust_clk – частота внешней интерфейсной части BPAM, соединенной с коммутатором отладочной логики уровня микросхемы.
- XTI32K/XTO32K – частота 32.768 КГц для тактирования таймера реального времени RTC;
- MFBSP0_LCLK, MFBSP1_LCLK – частоты внешних интерфейсов контроллеров MFBSP0, MFBSP1.

PLL подсистемы позволяет генерировать частоты в диапазоне от опорной частоты 27.456 МГц до частоты 3.185 ГГц.

В Таблица 5.1 дано описание тактовых частот подсистемы.

Таблица 5.1. Описание тактовых частот микросхемы

Название	Описание	Максимальные рабочие значения при НУ/WC	Ограничения	Источник
Внешние источники частоты				
clk_ref_service	Опорная частота PLL подсистемы. Общая частота подсистемы после сброса.	27.456 МГц		XTI_274 via

				top_clkgate reg
ust_clk	Частота работы интерфейса с SoC блока BPAM, частота работы отладочной логики UltraSoC микросхемы.	100 МГц		SoC
XTI32K	Частота 32КГц для тактирования таймера реального времени RTC	32.768 КГц		XTI32K/XTO32K
MFBSP0_LCLK	Частота внешнего интерфейса контроллера MFBSP0	100 МГц		MFBSP0_LCLK
MFBSP1_LCLK	Частота внешнего интерфейса контроллера MFBSP1	100 МГц		MFBSP1_LCLK
Внутренние частоты				
clk_apb[1)]	Частота работы системной части подсистемы – блоков PLL, URB, URG, UCG, части логики коммутатора в виде шины apbys и контроллеров, связанных с этой шиной.	100.7 / 100.7 МГц	Может относиться к clk_core как 1:1, 1:2, 1:3 …	UCG1 ch0
clk_core	Частота портов dev, ddr, sys подсистемы. Частота коммутационной логики, связанной с этими портами и основной логики коммутатора подсистемы	604 / 604 МГц		UCG1 ch1
clk_qspi0	Частота работы ядра блока QSPI0	604 / 604 МГц	Равна clk_core	UCG1 ch2
clk_bpam	Частота работы ядра блока BPAM	604 / 604 МГц	Равна clk_core	UCG1 ch3
clk_risc0	Частота работы RISC0	604 / 604 МГц	Равна clk_core	UCG1 ch4
clk_mfbsp0	Частота работы MFBSP0	100.7 / 100.7 МГц	Равна clk_apb	UCG1 ch5
clk_mfbsp1	Частота работы MFBSP1	100.7 / 100.7 МГц	Равна clk_apb	UCG1 ch6
clk_mailbox0	Частота работы MAILBOX0	100.7 / 100.7 МГц	Равна clk_apb	UCG1 ch7
clk_pvtctr	Частота работы PVT CTR	100.7 / 100.7 МГц	Равна clk_apb	UCG1 ch8
clk_i2c4	Частота работы ядра блока I2C4	100.7 / 100.7 МГц	Равна clk_apb	UCG1 ch9
clk_trng	Частота работы ядра блока TRNG	100.7 / 100.7 МГц	Равна clk_apb	UCG1 ch10
clk_spiotp	Частота работы ядра блока SPI_OTP	100.7 / 100.7 МГц	Равна clk_apb	UCG1 ch11
clk_i2c4_ext	Частота работы внешнего интерфейса I2C4 в режиме мастер	100.7 / 100.7 МГц		UCG1 ch12
clk_qspi0_ext	Частота работы внешнего интерфейса QSPI0 в режиме мастер	100.7 / 100.7 МГц		UCG1 ch13
clkout_ext	Частота, подающаяся на вывод CLKOUT микросхемы	100.7 / 100.7 МГц		UCG1 ch14
risc0_tck_ucg	Частота, подающаяся на ввод TCK RISC0 для доступа UltraSoC к OnCD	50.3 / 50.3 МГц		UCG1 ch15

В состоянии после сброса каналы формирования частот 3, 5 – 12 и 14, 15 UCG1 выключены и частоты не сконфигурированы, каналы 0, 1, 2, 4, 13 включены и работают по умолчанию на опорной частоте clk_ref_service.

Ограничения на соотношение частот перечисленные в Таблица 5.1 должны соблюдаться при настройке UCG1. Несоблюдение может привести к неработоспособности микросхемы. Обязательно выполнение процедуры синхронизации каналов, для которых есть требования по соотношению частот (п. 43.3.7 «Синхронизация каналов»).

[1)] Примечание:

Для блока OTP памяти существует ограничение по частоте чтения в 25 МГц, соответственно в приложениях, использующих OTP память, частота clk_apb ограничена

сверху значением в 25 МГц. Другие частоты подсистемы, зависящие от clk_abp, должны быть настроены соответствующе.

5.3 Логика сброса

Логика сброса служебной подсистемы отвечает, как за формирование общих сигналов сброса для остальных подсистем микросхемы, так и за формирование внутренних сигналов сброса для блоков подсистемы.

5.3.1 Глобальный сброс микросхемы

Источниками глобального сброса микросхемы являются:

- NRST_PON – сигнал сброса микросхемы с вывода NRST_PON, дополнительно сбрасывает RTC;
- NRST_WRM – сигнал сброса микросхемы с вывода NRST_WRM;
- rstn_wdt0 – сигнал сброса, вырабатываемый сторожевым таймером WDT0.

По событию глобального сброса происходит сброс всех подсистем и всей логики микросхемы.

5.3.2 Генерация сброса для подсистем микросхемы

Служебная подсистема содержит регистры PPU для управления состоянием остальных подсистем микросхемы, в том числе с их помощью могут формироваться общие сигналы сброса для подсистем. Схема формирования сброса для подсистемы дана на рисунке ниже.

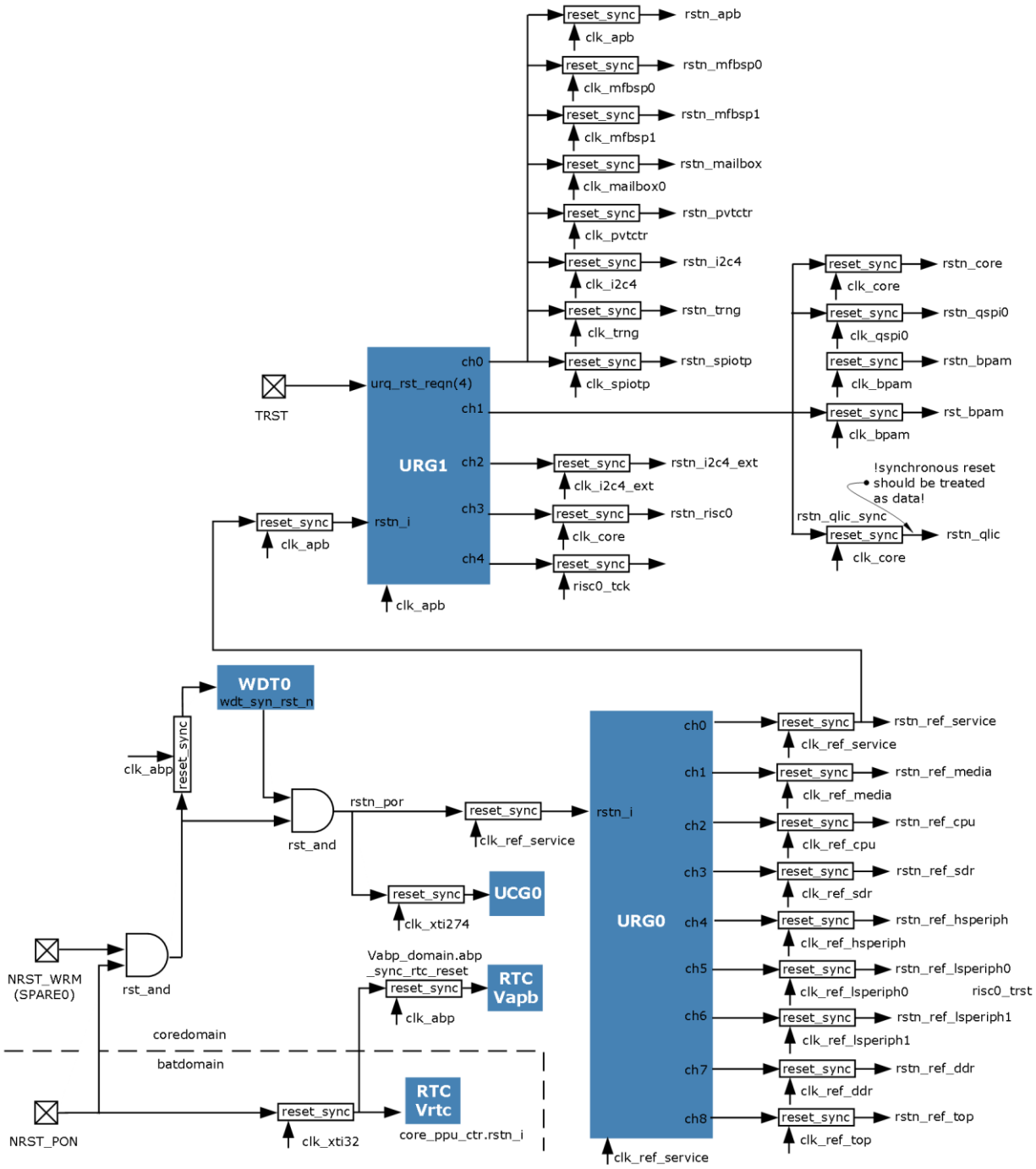


Рисунок 5.3. Схема генерации сброса в подсистеме

В таблице показаны источники общих сигналов сброса для подсистем и причины по которым происходит он происходит. Таблица использует следующую нотацию:

- Да – сброс является прямым следствием соблюдения условия;
- Нет – условие не приводит к сбросу;
- Р – условие приводит к сбросу не напрямую, а через сброс регистров РР, который в свою очередь инициирует процедуру сброса.

Перед выводом из сброса любой подсистемы (кроме top), необходимо перевести в xti_bypass частоты всех подключенных к подсистеме портов системного интерконнекта. Более подробную информацию можно получить в описании соответствующего подсистеме регистра *policy.

Таблица 5.2. Источники общих сигналов сброса подсистем

	Регистры _ppolicy												Внешние источники		
Компонент или сигнал сброса компонента	**CPU PP OFF**	**CPU PP WARM_ RST**	**SDR PP OFF**	**SDR PP WARM_ RST**	**MEDIA PP OFF**	**MEDIA PP WARM_ RST**	**HSPERIPH PP WARM_ RST**	**LSPERIPH0 PP WARM_ RST**	**LSPERIPH1 PP WARM_ RST**	**DDR PP WARM_ RST**	**TOP PP WARM _RST**	**CORE PP OFF**	**NRST_ PON**	**NRST_ WRM**	**rstn_wdt 0**
cpu_subs	Да	Да	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	*	Да	Да	Да
sdr_subs	Нет	Нет	Да	Да	Нет	Нет	Нет	Нет	Нет	Нет	Нет	*	Да	Да	Да
media_subs	Нет	Нет	Нет	Нет	Да	Да	Нет	Нет	Нет	Нет	Нет	*	Да	Да	Да
hsperiph_subs	Нет	Нет	Нет	Нет	Нет	Нет	Да	Нет	Нет	Нет	Нет	Да	Да	Да	Да
lsperiph0_subs	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Да	Нет	Нет	Нет	Да	Да	Да	Да
lsperiph1_subs	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Да	Нет	Нет	Да	Да	Да	Да
ddr_subs	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Да	Нет	Да	Да	Да	Да
top_subs	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Да	Да	Да	Да	Да
service_subs*	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Да	Да	Да	Да
service_subs/WDT0	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Да	Да	Да	Нет
service_subs/RTC	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Нет	Да	Нет	Нет

* кроме WDT0 и RTC

5.3.3 Внутренняя генерация сброса служебной подсистемы

Различные компоненты подсистемы могут сбрасываться по следующим причинам:

- Глобальный сброс микросхемы;
- Регистры PPU URB подсистемы;
- Для блока BPAM внешний вход подсистемы ust_rst – общий сигнал сброса отладочной логики;
- Для блока отладки OnCD RISC0 - JTAG вход микросхемы TRSTN.

В Таблица 5.3 показаны внутренние источники сброса и причины по которым происходит их сброс. Таблица использует следующую нотацию:

- Да – сброс является прямым следствием соблюдения условия;
- Нет – условие не приводит к сбросу;
- P – условие приводит к сбросу не напрямую, а через сброс регистров PP, который в свою очередь инициирует процедуру сброса.

Таблица 5.3. Источники сброса в подсистеме

Компонент или сигнал сброса компонента	RISC0 PP OFF	RISC0 PP WARM_RST	rstn_ref	rstn_wdt0	NRST_PON	JTAG TRSTN
RISC0	Да	Да	Да	Да	Да	Нет
RISC0.OnCD	Нет	Нет	Да	Да	Да	Да
RTC	Нет	Нет	Нет	Нет	Да	Нет
Остальная логика подсистемы	Нет	Нет	Да	Да	Да	Нет

5.3.4 Старт ядра RISC0 и начальная загрузка микросхемы после сброса

После сброса микросхемы включается служебная подсистема, а состояние остальных подсистем определяется значением по сбросу регистров _ppolicy. В данной конфигурации подсистемы начальные значения этих регистров установлены так, как показано в Таблица 5.4.

Таблица 5.4. Начальное состояние регистров _ppolicy подсистемы

Регистр	Подсистема или компонент	Состояние после сброса
cpu_ppolicy	Домен cpu	0, выключено
sdr_ppolicy	Домен sdr	0, выключено
media_ppolicy	Домен media	0, выключено
core_ppolicy	Домен core	‘b10000, ON
hsperiph_subs_ppolicy	hsperiph	‘b10000, ON
lsperiph0_subs_ppolicy	lsperiph0	0, выключено
lsperiph1_subs_ppolicy	lsperiph1	0, выключено
ddr_subs_ppolicy	ddr	0, выключено
top_subs_ppolicy	top	‘b10000, ON
risc0_ppolicy	RISC0	‘b10000, ON

Старт ядра RISC0 после включения микросхемы происходит автоматически с адреса 0x00_4000_0000 (QSPI0) при регистре boot == 'b000, либо с адреса 0x00_1fc00_0000 (RISC0 ROM) при других значениях регистра boot. Список возможных источников начальной загрузки приведен в Таблица 5.5.

Таблица 5.5. Источники начальной загрузки

Значение на выводах или fuse0 полях boot2,boot1,boot0	Источник
‘b000	загрузка напрямую через QSPI0
‘b001	загрузка через ROM RISC0/QSPI0
‘b010	загрузка через ROM RISC0/MFBSP0
‘b011	загрузка через ROM RISC0/UART0
‘b100	загрузка через ROM RISC0/SDMMC0
‘b101	загрузка через ROM RISC0 с быстрым стартом CPU из QSPI1
‘b110	резерв
‘b111	режим noBoot (RISC0 не загружается в ожидании сеанса отладки)

Рекомендуется следующий общий порядок старта для ядра RISC0 после сброса микросхемы:

- Загрузка в CRAM RISC0 и авторизация кода вторичного загрузчика для RISC0;
    - Далее исполнение кода вторичного загрузчика;
- Общая настойка микросхемы;
    - Включение PLL на требуемую частоту работы PLL;

- Настройка и включение через UCG1 частот блоков подсистемы;
- Включение доменов и подсистем микросхемы через регистры _ppolicy;
- Включение частот для подсистем микросхемы через регистр top_clkgate;

- Настройка подсистем микросхемы;
  - Включение и настройка частот остальных подсистем микросхемы через их регистры;
  - Настройка стартовых параметров подсистем микросхемы;
- Загрузка и авторизация кодов загрузчиков для ядер CPU, DSP0, DSP1, RISC1;
- Снятие сброса ядер и их старт через регистры _ppolicy подсистем.

5.3.5 Условия записи в регистры управления сбросом

При выводе блока из состояния сброса (как системного, так и программного) необходимо обеспечить на нем частоту сигнала, не превышающую XTI. Рекомендуемый метод реализации: перевод соответствующего канала UCG в bypass mode.

5.4 Обработка прерываний

В качестве контроллера прерываний в подсистеме используется блок QLIC0. Отображение внутренних прерываний микросхемы на QLIC0 показано в Раздел 3.5.2 данного руководства. Целевые сигналы для обработки прерываний, на которых возможно отображение внутренних прерываний в блоке QLIC0, показаны в Таблица 5.6.

Таблица 5.6. Цели для прерываний QLIC0

Номер	Целевое прерывание
0	RISC0 IP[2]
1	RISC0 IP[3]
2	RISC0 IP[4]
3	RISC0 IP[5]
4	QLIC0 irqreq / PPI28 CPU0
5	QLIC0 irqreq / PPI28 CPU1
6	QLIC0 irqreq / PPI28 CPU2
7	QLIC0 irqreq / PPI28 CPU3
8	QLIC0 irqreq / QLIC1 inq input
9	
10	Вывод микросхемы EXTINT1
11	

В служебной подсистеме нет устройств, формирующих немаскируемое прерывание (NMI) для RISC0, соответственно это исключение никогда не возникает.

5.5 Программная модель и регистры подсистемы

В Таблица 5.7 дана карта внутренней памяти подсистемы, доступная через ее порт sys.

Таблица 5.7. Карта внутренней памяти подсистемы

Стартовый адрес	Размер области	Описание
0x00_1efd_0000	16 Кбайт	Mailbox0 FIFO0 – FIFO3 (non-secure)
0x00_1efd_4000	8 Кбайт	Mailbox0 FIFO4 – FIFO5 (secure)
0x00_1efd_6000	8 Кбайт	Mailbox0 FIFO6 – FIFO7 (sdr)
0x00_1efd_8000	4 Кбайта	Mailbox0 SOFT_RESET
0x00_1efd_9000	-	Резерв
0x00_1f00_0000	64 Кбайт	Регистры URB
0x00_1f01_0000	4 Кбайта	Резерв
0x00_1f01_1000	-	Резерв
0x00_1f02_0000	4 Кбайта	Регистры UCG1
0x00_1f02_1000	-	Резерв
0x00_1f03_0000	512 байт	Память OTP
0x00_1f03_0200	-	Резерв
0x00_1f04_0000	64 Кбайта	Регистры SPI OTP
0x00_1f05_0000	64 Кбайта	Регистры PVT CTR
0x00_1f06_0000	64 Кбайта	Регистры RTC
0x00_1f07_0000	64 Кбайта	Регистры TRNG
0x00_1f08_0000	64 Кбайта	Регистры WDT0
0x00_1f09_0000	64 Кбайта	Регистры I2C4
0x00_1f0a_0000	64 Кбайта	Регистры MFBSP0
0x00_1f0b_0000	64 Кбайта	Регистры MFBSP1
0x00_1f0c_0000	-	Резерв
0x00_1fa0_0000	32 Кбайта	Память CRAM RISC0*
0x00_1fa0_8000	-	Резерв
0x00_1fc0_0000	64 Кбайта	Память ROM RISC0 (доступна только для RISC0)
0x00_1fc1_0000	-	Резерв
0x00_1fd0_0000	16 Кбайт	Регистры RISC0 QDMA
0x00_1fd0_4000	16 Кбайт	Регистры RISC0 VMMU

Стартовый адрес	Размер области	Описание
0x00_1fd0_8000	16 Кбайт	Регистры RISC0 CSR
0x00_1fd0_c000	16 Кбайт	Регистры RISC0 OnCD
0x00_1fd1_0000	-	Резерв
0x00_1fe0_0000	1 Мбайт	Регистры QLIC0
0x00_1ff0_0000	64 Кбайта	Регистры QSPI0
0x00_1ff1_0000	-	Резерв
0x00_4000_0000	1 Мбайт	Память QSPI0 XiP (только чтение)
0x00_4000_0000	-	Резерв

**внешние байтовые обращение к данной памяти не поддерживаются*

Области регистров RISC0 OnCD, RISC0 CSR, RISC0 VMMU, RISC0 QDMA, доступны также по интерфейсу ust через компонент отладки BPAM.

Далее дано описание регистров URB, остальные регистры внутренней памяти подсистемы описаны в главах, посвященных соответствующим блокам.

5.5.1 Разграничение доступа для внутренних транзакций в подсистеме

Для коммутаторов подсистемы введен дополнительный функционал по разграничению доступа для внутренних транзакций. См. регистры axicore_internal_priv и axisys_internal_priv. Изменять политику доступа с помощью этих регистров допускается только во время настройки микросхемы при начальной загрузке. Привилегии по доступу статически распределены среди внутренних мастеров подсистемы следующим образом:

- привилегированный доступ – мастер RISC0 (vcpu);
- не привилегированный доступ – мастера DMA MFBSP0 и MFBSP1, DMA RISC0 (qdma), внешний;

Уровень привилегированности внешнего порта подсистемы sys меняется динамически согласно битам AxPROT[1] транзакций.

В случае попытки доступа от непривилегированных мастеров в область с привилегированным доступом, на коммутаторе подсистемы генерируется ответное сообщение об ошибке.

5.5.2 Область OTP памяти

Область OTP памяти представляет собой накристальную энергонезависимую память из 128 32-х разрядных слов с дополнительными ECC битами. Память доступна по чтению пословно напрямую. Запись осуществляется с помощью контроллера SPI_OTP, при этом перезапись возможна только в биты с текущим значением 0. В Таблица 5.8 показано использование OTP памяти.

Таблица 5.8. Область OTP памяти

Номер слова	Размер (в словах)	Название	Описание
0	1	fuse1	
1	1	fuse0	
2	30	BootROM specific	Используются BootROM
32	1	fuse1 redundant	
33	1	fuse0 redundant	
34	2	BootROM specific	Используется BootROM
36	11	–	Резерв
47	17	Firmware defined lockable	Определяется прошивкой устройства, есть возможность блокировки дозаписи (OTP.fuse1.lock_fw)
64	64	Firmware defined	Определяется прошивкой устройства

5.5.3 Регистры URB подсистемы

Перечень регистров service_subs_urb приведен в следующей таблице:

Таблица 5.9. Перечень программно-доступных регистров service_subs_urb

Смещение	Условное обозначение	Название регистра	Исходное состояние	Тип доступа
0x0000	cpu_ppolicy	Регистр установки состояния подсистемы cpu_subs и домена cpu	0x0	RW
0x0004	cpu_pstatus	Регистр статуса состояния подсистемы cpu_subs и домена cpu	0x0	RO
0x0008	sdr_ppolicy	Регистр установки состояния подсистемы sdr_subs и домена sdr	0x0	RW
0x000c	sdr_pstatus	Регистр статуса состояния подсистемы sdr_subs и домена sdr	0x0	RO
0x0010	media_ppolicy	Регистр установки состояния подсистемы media_subs и домена media	0x0	RW
0x0014	media_pstatus	Регистр статуса состояния подсистемы media_subs и домена media	0x0	RO
0x0018	core_ppolicy	Регистр установки состояния домена core	0x0	RW
0x001c	core_pstatus	Регистр статуса состояния домена core	0x0	RO

0x0020	hsperiph_subs_ppolicy	Регистр установки состояния подсистемы hsperiph_subs	0x0	RW
0x0024	hsperiph_subs_pstatus	Регистр статуса состояния подсистемы hsperiph_subs	0x0	RO
0x0028	lsperiph0_subs_ppolicy	Регистр установки состояния подсистемы lsperiph0_subs	0x0	RW
0x002c	lsperiph0_subs_pstatus	Регистр статуса состояния подсистемы lsperiph0_subs	0x0	RO
0x0030	lsperiph1_subs_ppolicy	Регистр установки состояния подсистемы lsperiph1_subs	0x0	RW
0x0034	lsperiph1_subs_pstatus	Регистр статуса состояния подсистемы lsperiph1_subs	0x0	RO
0x0038	ddr_subs_ppolicy	Регистр установки состояния подсистемы ddr_subs	0x0	RW
0x003c	ddr_subs_pstatus	Регистр статуса состояния подсистемы ddr_subs	0x0	RO
0x0040	top_subs_ppolicy	Регистр установки состояния подсистемы верхнего уровня top_subs	0x0	RW
0x0044	top_subs_pstatus	Регистр статуса состояния подсистемы верхнего уровня ddr_subs	0x0	RO
0x0048	risc0_ppolicy	Регистр установки состояния risc0	0x0	RW
0x004c	risc0_pstatus	Регистр статуса состояния risc0	0x0	RO
0x1000	service_subs_pllcnfg	Регистр управления PLL подсистемы	0x0	RW
0x1004	service_subs_plldiag	Регистр диагностики PLL подсистемы	0x0	RW
0x1008	top_clkgate	Регистр включения опорных частот подсистем	0x111	RW
0x2000	boot	Регистр установок начальной загрузки	0x0	RO
0x2004	xip_en_req	Регистр включения режима XIP QSPI0	0x0	RW
0x2008	xip_en_out	Регистр статуса режима XIP QSPI0	0x0	RO
0x200c	xip_cfg	Регистр конфигурации режима XIP QSPI0	0x0	RW
0x2010	axicore_dlock	Регистр отладочной информации по коммутатору axicore	0x0	RO
0x2014	axisys_dlock	Регистр отладочной информации по коммутатору axisys	0x0	RO
0x2018	tscount_high	Регистр timestamp счетчика микросхемы, старшая часть	0x0	RO
0x201c	tscount_low	Регистр timestamp счетчика микросхемы, младшая часть	0x0	RO
0x2020	extint_cfg	Регистр конфигурации выходов extint	0x0	RW
0x2024	pads_ren	Регистр включения резисторов подтяжки для выводов подсистемы	0x0	RW

0x3000	fuse0	Регистр fuse0 микросхемы	0x0	RO
0x3004	fuse1	Регистр fuse1 микросхемы	0x0	RO
0x3008	otp_mode	Режим работы otp	0x0	RW
0x300c	otp_flag	Регистр сигнала FLAG otp	0x0	RO
0x3010	fuse0_ecc	Регистр проверки корректности fuse0	0x0	RO
0x3014	fuse1_ecc	Регистр проверки корректности fuse1	0x0	RO
0x3018	otp_ecc	Регистр проверки корректности данных читаемых из otp	0x0	RO
0x3020	axicore_internal_priv	Регистр разграничения доступа для внутренних компонентов подсистемы на уровне коммутатора axicore	0x0	RW
0x3024	axisys_internal_priv	Регистр разграничения доступа для внутренних компонентов подсистемы на уровне коммутатора axicore	0x0	RW
0x3030	tp_dbgen	Регистр контроля отладочного доступа для DPM_TP	0x0	RW
0x3034	sdr_dbgen	Регистр контроля отладочного доступа для DPM_SDR	0x0	RW
0x3038	sp_dbgen	Регистр контроля отладочного доступа для DPM_SP	0x0	RW
0x303C	s_dbgen	Регистр контроля отладочного доступа для DPM_SP, DPM_SU, DPM_NSP, DPM_NSU	0x0	RW
0x3040	ust_dbgen	Регистр контроля отладочного доступа для компонентов отладки UltraSoC	0x0	RW

5.5.4 Описание регистра cpu_ppolicy

Формат регистра cpu_ppolicy приведен в следующей таблице:

Таблица 5.10. Поля регистра cpu_ppolicy

Диапазон	Название	Описание	Сброс
31:5	-	Резерв	0
4:0	policy	Установка состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; 'b00001 PP_OFF - выключено; другое - резерв/выключено Перед переводом регистра в состояние PP_ON, необходимо перевести в XTI_BYPASS следующие каналы UCG0 уровня top: ○ ddr_cpu (clk_out[4]) ○ cpu_acp (clk_out[5]) ○ axi_coh_comm (clk_out[7])	0

Диапазон	Название	Описание	Сброс
		и каналы UCG1 уровня top: ○ axi_slow_comm (clk_out[0]) ○ axi_fast_comm (clk_out[2]) Для этого необходимо установить в 1 соответствующие биты регистров UCG_BP_CTR_REG блоков UCG0 и UCG1. Bypass указанных выше каналов можно выключить сразу после перехода регистра cpu_pstatus.pstatus в состояние PP_ON.	

5.5.5 Описание регистра cpu_pstatus

Формат регистра cpu_pstatus приведен в следующей таблице:

Таблица 5.11. Поля регистра cpu_pstatus

Диапазон	Название	Описание	Сброс
31:9	-	Резерв	0
8	memretspt	Состояние MEM_RET поддерживается в установках для этого домена	0
7:6	-	Резерв	0
5	emulated	Значение 1 указывает что состояние OFF эмулируется для нужд отладки	0
4:0	pstatus	Поле отражает текущую активную установку состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; 'b00001 PP_OFF - выключено; другое - резерв/выключено	0

5.5.6 Описание регистра sdr_ppolicy

Формат регистра sdr_ppolicy приведен в следующей таблице:

Таблица 5.12. Поля регистра sdr_ppolicy

Диапазон	Название	Описание	Сброс
31:5	-	Резерв	0

Диапазон	Название	Описание	Сброс
4:0	policy	Установка состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; 'b00001 PP_OFF - выключено; другое - резерв/выключено Перед переводом регистра в состояние PP_ON, необходимо перевести в XTI_BYPASS следующие каналы UCG1 уровня top: o axi_slow_comm (clk_out[0]) o axi_fast_comm (clk_out[2]) o ddr_sdr_dsp (clk_out[4]) o ddr_sdr_pice (clk_out[5]) Для этого необходимо установить в 1 соответствующие биты регистра UCG_BP_CTR_REG UCG1. Bypass указанных выше каналов можно выключить сразу после перехода регистра sdr_pstatus.pstatus в состояние PP_ON.	0

5.5.7 Описание регистра sdr_pstatus

Формат регистра sdr_pstatus приведен в следующей таблице:

Таблица 5.13. Поля регистра sdr_pstatus

Диапазон	Название	Описание	Сброс
31:9	-	Резерв	0
8	memretspt	Состояние MEM_RET поддерживается в установках для этого домена	0
7:6	-	Резерв	0
5	emulated	Значение 1 указывает что состояние OFF эмулируется для нужд отладки	0
4:0	pstatus	Поле отражает текущую активную установку состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; 'b00001 PP_OFF - выключено; другое - резерв/выключено	0

5.5.8 Описание регистра media_ppolicy

Формат регистра media_ppolicy приведен в следующей таблице:

Таблица 5.14. Поля регистра media_ppolicy

Диапазон	Название	Описание	Сброс
31:5	-	Резерв	0

Диапазон	Название	Описание	Сброс
4:0	policy	Установка состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; 'b00001 PP_OFF - выключено; другое - резерв/выключено Перед переводом регистра в состояние PP_ON, необходимо перевести в XTI_BYPASS следующие каналы UCG0 уровня top: o ddr_dp (clk_out[0]) o ddr_vpu (clk_out[1]) o ddr_gpu (clk_out[2]) o ddr_isp (clk_out[3]) и каналы UCG1 уровня top: o axi_slow_comm (clk_out[0]) Для этого необходимо установить в 1 соответствующие биты регистров UCG_BP_CTR_REG блоков UCG0 и UCG1. Bypass указанных выше каналов можно выключить сразу после перехода регистра media_pstatus.pstatus в состояние PP_ON.	0

5.5.9 Описание регистра media_pstatus

Формат регистра media_pstatus приведен в следующей таблице:

Таблица 5.15. Поля регистра media_pstatus

Диапазон	Название	Описание	Сброс
31:9	-	Резерв	0
8	memretspt	Состояние MEM_RET поддерживается в установках для этого домена	0
7:6	-	Резерв	0
5	emulated	Значение 1 указывает что состояние OFF эмулируется для нужд отладки	0
4:0	pstatus	Поле отражает текущую активную установку состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; 'b00001 PP_OFF - выключено; другое - резерв/выключено	0

5.5.10 Описание регистра core_ppolicy

Формат регистра core_ppolicy приведен в следующей таблице:

Таблица 5.16. Поля регистра core_ppolicy

Диапазон	Название	Описание	Сброс
31:5	-	Резерв	0

Диапазон	Название	Описание	Сброс
4:0	policy	Установка состояния: 'b10000 PP_ON - включено; 'b00001 PP_OFF - выключено; другое - резерв/выключено	'b10000

5.5.11 Описание регистра core_pstatus

Формат регистра core_pstatus приведен в следующей таблице:

Таблица 5.17. Поля регистра core_pstatus

Диапазон	Название	Описание	Сброс
31:9	-	Резерв	0
8	memretspt	Состояние MEM_RET поддерживается в установках для этого домена	0
7:6	-	Резерв	0
5	emulated	Значение 1 указывает что состояние OFF эмулируется для нужд отладки	0
4:0	pstatus	Поле отражает текущую активную установку состояния: 'b10000 PP_ON - включено; 'b00001 PP_OFF - выключено; другое - резерв/выключено	0

5.5.12 Описание регистра hsperiph_subs_ppolicy

Формат регистра hsperiph_subs_ppolicy приведен в следующей таблице:

Таблица 5.18. Поля регистра hsperiph_subs_ppolicy

Диапазон	Название	Описание	Сброс
31:5	-	Резерв	0
4:0	policy	Установка состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; 'b00001 PP_OFF - выключено; другое - резерв/выключено Перед переводом регистра в состояние PP_ON, необходимо перевести в XTI_BYPASS следующие каналы UCG1 уровня top: o axi_slow_comm (clk_out[0]) o axi_fast_comm (clk_out[2]) o ddr_hsperiph (clk_out[8]) Для этого необходимо установить в 1 соответствующие биты регистров UCG_BP_CTR_REG блока UCG1. Bypass указанных выше каналов можно выключить сразу после перехода регистра hsperiph_subs_pstatus.pstatus в состояние PP_ON.	'b10000

5.5.13 Описание регистра hsperiph_subs_pstatus

Формат регистра hsperiph_subs_pstatus приведен в следующей таблице:

Таблица 5.19. Поля регистра hsperiph_subs_pstatus

Диапазон	Название	Описание	Сброс
31:9	-	Резерв	0
8	memretspt	Состояние MEM_RET поддерживается в установках для этого домена	0
7:6	-	Резерв	0
5	emulated	Значение 1 указывает что состояние OFF эмулируется для нужд отладки	0
4:0	pstatus	Поле отражает текущую активную установку состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; 'b00001 PP_OFF - выключено; другое - резерв/выключено	0

5.5.14 Описание регистра lsperiph0_subs_ppolicy

Формат регистра lsperiph0_subs_ppolicy приведен в следующей таблице:

Таблица 5.20. Поля регистра lsperiph0_subs_ppolicy

Диапазон	Название	Описание	Сброс
31:5	-	Резерв	0
4:0	policy	Установка состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; 'b00001 PP_OFF - выключено; другое - резерв/выключено Перед переводом регистра в состояние PP_ON, необходимо перевести в XTI_BYPASS следующие каналы UCG0 уровня top: o ddr_lsperiph0 (clk_out[6]) и каналы UCG1 уровня top: o axi_slow_comm (clk_out[0]) Для этого необходимо установить в 1 соответствующие биты регистров UCG_BP_CTR_REG блоков UCG0 и UCG1. Bypass указанных выше каналов можно выключить сразу после перехода регистра lsperiph0_subs_pstatus.pstatus в состояние PP_ON.	0

5.5.15 Описание регистра lsperiph0_subs_pstatus

Формат регистра lsperiph0_subs_pstatus приведен в следующей таблице:

Таблица 5.21. Поля регистра lsperiph0_subs_pstatus

Диапазон	Название	Описание	Сброс
31:9	-	Резерв	0
8	memretspt	Состояние MEM_RET поддерживается в установках для этого домена	0
7:6	-	Резерв	0
5	emulated	Значение 1 указывает что состояние OFF эмулируется для нужд отладки	0
4:0	pstatus	Поле отражает текущую активную установку состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; 'b00001 PP_OFF - выключено; другое - резерв/выключено	0

5.5.16 Описание регистра lsperiph1_subs_ppolicy

Формат регистра lsperiph1_subs_ppolicy приведен в следующей таблице:

Таблица 5.22. Поля регистра lsperiph1_subs_ppolicy

Диапазон	Название	Описание	Сброс
31:5	-	Резерв	0
4:0	Policy	Установка состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; 'b00001 PP_OFF - выключено; другое - резерв/выключено Перед переводом регистра в состояние PP_ON, необходимо перевести в XTI_BYPASS следующие каналы UCG1 уровня top: o axi_slow_comm (clk_out[0]) o ddr_lsperiph1(clk_out[6]) Для этого необходимо установить в 1 соответствующие биты регистров UCG_BP_CTR_REG блока UCG1. Bypass указанных выше каналов можно выключить сразу после перехода регистра lsperiph1_subs_pstatus_pstatus.pstatus в состояние PP_ON.	0

5.5.17 Описание регистра lsperiph1_subs_pstatus

Формат регистра lsperiph1_subs_pstatus приведен в следующей таблице:

Таблица 5.23. Поля регистра lsperiph1_subs_pstatus

Диапазон	Название	Описание	Сброс
31:9	-	Резерв	0
8	memretspt	Состояние MEM_RET поддерживается в установках для этого домена	0
7:6	-	Резерв	0

Диапазон	Название	Описание	Сброс
5	emulated	Значение 1 указывает что состояние OFF эмулируется для нужд отладки	0
4:0	pstatus	Поле отражает текущую активную установку состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; 'b00001 PP_OFF - выключено; другое - резерв/выключено	0

5.5.18 Описание регистра ddr_subs_ppolicy

Формат регистра ddr_subs_ppolicy приведен в следующей таблице:

Таблица 5.24. Поля регистра ddr_subs_ppolicy

Диапазон	Название	Описание	Сброс
31:5	-	Резерв	0
4:0	policy	Установка состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; 'b00001 PP_OFF - выключено; другое - резерв/выключено Перед переводом регистра в состояние PP_ON, необходимо перевести в XTI_BYPASS следующие каналы UCG0 уровня top: o ddr_dp (clk_out[0]) o ddr_vpu (clk_out[1]) o ddr_gpu (clk_out[2]) o ddr_isp (clk_out[3]) o ddr_cpu (clk_out[4]) o ddr_lsperiph0 (clk_out[6]) и каналы UCG1 уровня top: o axi_slow_comm (clk_out[0]) o ddr_sdr_dsp (clk_out[4]) o ddr_sdr_pcie (clk_out[5]) o ddr_lsperiph1 (clk_out[6]) o ddr_service (clk_out[7]) o ddr_hsperiph (clk_out[8]) Для этого необходимо установить в 1 соответствующие биты регистров UCG_BP_CTR_REG блоков UCG0 и UCG1. Bypass указанных выше каналов можно выключить сразу после перехода регистра ddr_subs_pstatus.pstatus в состояние PP_ON.	0

5.5.19 Описание регистра ddr_subs_pstatus

Формат регистра ddr_subs_pstatus приведен в следующей таблице:

Таблица 5.25. Поля регистра ddr_subs_pstatus

Диапазон	Название	Описание	Сброс
31:9	-	Резерв	0
8	memretspt	Состояние MEM_RET поддерживается в установках для этого домена	0
7:6	-	Резерв	0
5	emulated	Значение 1 указывает что состояние OFF эмулируется для нужд отладки	0
4:0	pstatus	Поле отражает текущую активную установку состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; 'b00001 PP_OFF - выключено; другое - резерв/выключено	0

5.5.20 Описание регистра top_subs_ppolicy

Формат регистра top_subs_ppolicy приведен в следующей таблице:

Таблица 5.26. Поля регистра top_subs_ppolicy

Диапазон	Название	Описание	Сброс
31:5	-	Резерв	0
4:0	policy	Установка состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; 'b00001 PP_OFF - выключено; другое - резерв/выключено	'b10000

5.5.21 Описание регистра top_subs_pstatus

Формат регистра top_subs_pstatus приведен в следующей таблице:

Таблица 5.27. Поля регистра top_subs_pstatus

Диапазон	Название	Описание	Сброс
31:9	-	Резерв	0
8	memretspt	Состояние MEM_RET поддерживается в установках для этого домена	0
7:6	-	Резерв	0
5	emulated	Значение 1 указывает что состояние OFF эмулируется для нужд отладки	0
4:0	pstatus	Поле отражает текущую активную установку состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; 'b00001 PP_OFF - выключено; другое - резерв/выключено	0

5.5.22 Описание регистра risc0_ppolicy

Формат регистра risc0_ppolicy приведен в следующей таблице:

Таблица 5.28. Поля регистра risc0_ppolicy

Диапазон	Название	Описание	Сброс
31:5	-	Резерв	0
4:0	policy	Установка состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; 'b00001 PP_OFF - выключено; другое - резерв/выключено	0

5.5.23 Описание регистра risc0_pstatus

Формат регистра risc0_pstatus приведен в следующей таблице:

Таблица 5.29. Поля регистра risc0_pstatus

Диапазон	Название	Описание	Сброс
31:9	-	Резерв	0
8	memretspt	Состояние MEM_RET поддерживается в установках для этого домена	0
7:6	-	Резерв	0
5	emulated	Значение 1 указывает что состояние OFF эмулируется для нужд отладки	0
4:0	pstatus	Поле отражает текущую активную установку состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; 'b00001 PP_OFF - выключено; другое - резерв/выключено	0

5.5.24 Описание регистра service_subs_pllcnfg

Формат регистра service_subs_pllcnfg приведен в следующей таблице:

Таблица 5.30. Поля регистра service_subs_pllcnfg

Диапазон	Название	Описание	Сброс
31	lock	Бит указывает окончание настройки PLL на новую заданную частоту работы. При начале процедуры настройки либо при выключенной PLL сбрасывается в 0.	0
30:27	nr_man	Коэффициент nr настройки pll	
26:14	nf_man	Коэффициент nf настройки pll	0
13:10	od_man	Коэффициент od настройки pll	0
9:9	man	При бите заданном в 1 и sel > 0 конфигурация pll определяется полям *_man данного регистра. Выходная частота определяется по формуле ref_clk * (nf_man + 1) / (nr_man + 1) / (od_man + 1). Поля *_man рекомендуется задавать исходя из значений генерируемым скриптом настройки pll. Значение выходной частоты не должно превышать 3600 МГц.	0
8:8	-	Резерв	0

Диапазон	Название	Описание	Сброс
7:0	sel	Определяет выходную частоту и режим работы PLL: 0x0 - опорная частота ref_clk, PLL выключена; 0x1 - ref_clk * 2; 0x2 - ref_clk * 3; 0x3 - ref_clk * 4; ... - ...; 0x73 - ref_clk * 116; 0x74 - ref_clk * 116; ... - ...; 0xFF - ref_clk * 116; При sel > 0 и man = 1, конфигурация pll определяется полями *_man данного регистра	0

5.5.25 Описание регистра service_subs_plldiag

Формат регистра service_subs_pllcnfg приведен в следующей таблице:

Таблица 5.31. Поля регистра service_subs_plldiag

Диапазон	Название	Описание	Сброс
31:5	-	Резерв	
4:4	fbslip	Диагностический выход pll - регистрирует ситуацию feedback cycle slip	0
3:3	rfslip	Диагностический выход pll - регистрирует ситуацию referency cycle slip	0
2:2	fasten	Диагностический вход pll - разрешает механизм fast lock	0
1:1	ensat	Диагностический вход pll - разрешает насыщение, в нормальном режиме - 1	1
0:0	test	Диагностический вход pll - bypass опорной частоты pll на выход при 1	0

5.5.26 Описание регистра top_clkgate

Формат регистра top_clkgate приведен в следующей таблице:

Таблица 5.32. Поля регистра top_clkgate

Диапазон	Название	Описание	Сброс
31:5	-	Резерв	
8:0	en	Разрешение опорной частоты для подсистем по битам: 0 - service 1 - media 2 - cpu 3 - sdr 4 - hsperiph 5 - lsperiph0 6 - lsperiph1 7 - ddr 8 - top interconnect	0x111

5.5.27 Описание регистра boot

Формат регистра boot приведен в следующей таблице:

Таблица 5.33. Поля регистра boot

Диапазон	Название	Описание	Сброс	Тип
31:4	-	Резерв	0	
3	bs_en	Бит, установленный в 1, означает необходимость верификации образа начального загрузчика	0	r
2:0	boot	Регистр отражает установки микросхемы на режим работы и источник начальной загрузки, заданые через выводы BOOT микросхемы, либо через биты FUSE: 'b000 - загрузка напрямую через QSPI0; 'b001 - загрузка через ROM RISC0/QSPI0; 'b010 - загрузка через ROM RISC0/MFBSP0; 'b011 - загрузка через ROM RISC0/UART0; 'b100 - загрузка через ROM RISC0/SDMMC0; 'b101 - загрузка через ROM RISC0 с быстрым стартом CPU из QSPI1; 'b110 - загрузка через ROM RISC0/SPI0; 'b111 - режим noBoot (RISC0 не загружается в ожидании сеанса отладки);	0	r

5.5.28 Описание регистра xip_en_req

Формат регистра xip_en_req приведен в следующей таблице:

Таблица 5.34. Поля регистра xip_en_req

Диапазон	Название	Описание	Сброс
31:1	-	Резерв	0
0	en	Для включения режим XIP установить бит в 1, обращения по адресам памяти QSPI0 и изменение регистра XIP_CFG запрещены до установки регистра XIP_EN_OUT в 1. Для отключения режима XIP установить бит в 0, обращения по адресам памяти QSPI0 и изменение регистра XIP_CFG запрещены до установки регистра XIP_EN_OUT в 0	1

5.5.29 Описание регистра xip_en_out

Формат регистра xip_en_out приведен в следующей таблице:

Таблица 5.35. Поля регистра xip_en_out

Диапазон	Название	Описание	Сброс
31:1	-	Резерв	0
0	status	Статус режима XIP, 0 - режим выключен, 1 - режим включен;	0

5.5.30 Описание регистра xip_cfg

Формат регистра xip_cfg приведен в следующей таблице:

Таблица 5.36. Поля регистра xip_cfg

Диапазон	Название	Описание	Сброс
31	-	Резерв	0
30	hpen	0 - режим High Performance отключен, 1 - режим High Performance включен	0
29	le32	0 - читаемые данные Big Endian, 1 - читаемы данные 32-битные Little Endian	1
28	addr4	4-байтный режим адресации. 0 - контроллер выдает 3 байта адреса, 1 - контроллер выдае 4 байта адреса.	0
27	cpol	Режим CPOL	0
26	cpha	Режим CPHA	0
25:24	hp_end_dummy	Задает количество тактов ожидания, необходимое для выхода из режима High Performance mode. Используется контроллером при выходе из XIP режима, чтобы убедиться что Flash память не осталась в режиме HP. Для режима Dual количество тактов ожидания равно данному значению умноженному на 4. Для режима Quad количество тактов ожидания равно данному значению умноженному на 2.	0
23:20	dummy_cycles	Количество тактов ожидания при приеме. Для режимов Fast Read, Fast Read Dual Output, Fast Read Quad Output количество тактов ожидания равно данному значению умноженному на 8. Для режима Fast Read Dual I/O количество тактов ожидания равно данному значению умноженному на 4 плюс 4. Для режима Fast Read Quad I/O количество тактов ожидания равно данному значению умноженному на 2 плюс 2.	0
19:16	ssen	Выбор активного SS QSPI0 4'b0001 - Slave Select 0, 8'b0010 - Slave Select 1, ... 8'b1000 - Slave Select 3. Другие значения зарезервированы.	1
15:8	hp_mode	Данный байт передается в подключенную Flash память, в контрольные регистры, и в зависимости от модели активирует опции производительнсти в режимах Dual I/O и Quad I/O.	0
7:0	cmd	Тип команды на чтение: 8'h03 - Read; 8'h0B - Fast Read; 8'h3B - Fast Dual Out Read; 8'h6B - Fast Quad Out Read; 8'hBB - Fast Dual I/O Read; 8'hEB - Fast Quad I/O Read; Другие значения зарезервированы	8'h03

5.5.31 Описание регистра axicore_dlock

Формат регистра axicore_dlock приведен в следующей таблице:

Таблица 5.37. Поля регистра axicore_dlock

Диапазон	Название	Описание	Сброс
31:7	-	Резерв	0
6	dlock_id	Показывает id для транзакции приведшей к зависанию коммутатора	0
5:4	dlock_mst	Показывает номер мастера для транзакции приведшей к зависанию коммутатора	0
3:1	dlock_slv	Показывает номер слейва для транзакции приведшей к зависанию коммутатора	0
0	dlock_wr	Показывает запись или чтение привело к зависанию коммутатора	0

5.5.32 Описание регистра axisys_dlock

Формат регистра axisys_dlock приведен в следующей таблице:

Таблица 5.38. Поля регистра axisys_dlock

Диапазон	Название	Описание	Сброс
31:8	-	Резерв	0
7:5	dlock_id	Показывает id для транзакции приведшей к зависанию коммутатора	0
4	dlock_mst	Показывает номер мастера для транзакции приведшей к зависанию коммутатора	0
3:1	dlock_slv	Показывает номер слейва для транзакции приведшей к зависанию коммутатора	0
0	dlock_wr	Показывает запись или чтение привело к зависанию коммутатора	0

5.5.33 Описание регистра tscount_high

Формат регистра tscount_high приведен в следующей таблице:

Таблица 5.39. Поля регистра tscount_high

Диапазон	Название	Описание	Сброс
31:0	ts_high	Старшая часть значения timestamp счетчика микросхемы	0

5.5.34 Описание регистра tscount_low

Формат регистра tscount_low приведен в следующей таблице:

Таблица 5.40. Поля регистра tscount_low

Диапазон	Название	Описание	Сброс
31:0	ts_high	Старшая часть значения timestamp счетчика микросхемы	0

5.5.35 Описание регистра extint_cfg

Формат регистра extint_cfg приведен в следующей таблице:

Таблица 5.41. Поля регистра extint_cfg

Диапазон	Название	Описание	Сброс
31:2	–	Резерв	?
1	outen	Управление направления вывода EXTINT1 микросхемы: 0 - вывод работает в режиме входа; 1 - вывод работает в режиме выхода.	0
0	–	Резерв	?

5.5.36 Описание регистра pads_ren

Формат регистра pads_ren приведен в следующей таблице:

Таблица 5.42. Поля регистра pads_ren

Диапазон	Название	Описание	Сброс
31:5	-	Резерв	0
4	mfbsp0_ren	1'b0 - подтяжка включена; 1'b1 - подтяжка выключена	0
3	mfbsp1_ren	1'b0 - подтяжка включена; 1'b1 - подтяжка выключена	0
2	i2c4_ren	1'b0 - подтяжка включена; 1'b1 - подтяжка выключена	0
1	qspi0_ren	1'b0 - подтяжка включена; 1'b1 - подтяжка выключена	0
0	sysio_ren	1'b0 - подтяжка включена; 1'b1 - подтяжка выключена	0

5.5.37 Описание регистра fuse0

Формат регистра fuse0 приведен в следующей таблице:

Таблица 5.43. Поля регистра fuse0

Диапазон	Название	Описание	Сброс
31	lock	Установка бита блокирует перезапись fuse1	0
30	-	Резерв	0
29	-	Резерв	0
28	-	Резерв	0
27	top_subs_disable	Отключает коммутаторы верхнего уровня	0
26	ddr_subs_disable	Отключает подсистему ddr	0
25	ls1_subs_disable	Отключает подсистему lsperiph1	0
24	ls0_subs_disable	Отключает подсистему lsperiph0	0
23	hs_subs_disable	Отключает подсистему hsperiph	0
22	media_subs_disable	Отключает подсистему media	0
21	sdr_subs_disable	Отключает подсистему sdr	0
20	cpu_subs_disable	Отключает подсистему cpu	0
19	scan_test_disable	Отключает возможность DFT scan тестирования	0
18	mbist_test_disable	Отключает возможнось MBIST тестирования	0
17	bsr_test_disable	Октлючает возможность BSR тестирования	0
16	ust_debug_disable	Отключает возможность отладки через UltraSoC	0
15	bringupdbg_disable	Установка в 1 запрещает отладку через порт JTAG процессора RISC0	0

Диапазон	Название	Описание	Сброс
14	sdrtosecure_disable	Установка в 1 запрещает изменение прав доступа с sdr на secure. После прожига этого бита признак securable становиться не применим к областям с доверенным доступом (sdr)	0
13	trustedtosecure_disable	Установка в 1 запрещает изменение прав доступа с trusted на secure. После прожига этого бита признак securable становиться не применим к областям с доверенным доступом (trusted)	0
12	dpm_enable	Установка в 1 включает механизм проверки для всех отладочных DPM при dpm_lock == 1	0
11	-	Резерв	0
10	dpm_lock_secure	Установка в 1 запрещает операцию unlock для механизмов защиты отладочного доступа DPM_SP	0
9	dpm_lock_sdr	Установка в 1 запрещает операцию unlock для механизмов защиты отладочного доступа DPM_SDR	0
8	dpm_lock_trusted	Установка в 1 запрещает операцию unlock для механизмов защиты отладочного доступа DPM_TP	0
7	bs_en_padoverride	Установка в 1 задает установку bs_en с бита bs_en регистра, а не с внешнего вывода BS_EN микросхемы	0
6	bs_en	Установка в 1 задает необходимость верификации образа загрузчика при начальной загрузке	0
5	vs_en_padoverride	Установка в 1 задает установку vs_en с бита vs_en регистра, а не с внешнего вывода VS_EN микросхемы	0
4	vs_en	Установка в 1 уравнивает уровни доступа non_secure, secure, trusted и sdr	0
3	boot_padsoverride	Установка в 1 задает установки загрузки из битов boot0-2, а не с внешних выводов BOOT микросхемы	0
2	boot2	Задает источник для начальной загрузки микросхемы	0
1	boot1	Задает источник для начальной загрузки микросхемы	0
0	boot0	Задает источник для начальной загрузки микросхемы	0

5.5.38 Описание регистра fuse1

Формат регистра fuse1 приведен в следующей таблице:

Таблица 5.44. Поля регистра fuse1

Диапазон	Название	Описание	Сброс
31:31	lock	Установка бита блокирует перезапись fuse0	
30:30	lock_fw	Установка бита блокирует от перезаписи область 0xBC-FF OTP памяти	
29:29	lock_bootrom	Установка бита блокирует от перезаписи область 0x08 - 0x43 OTP памяти	
28:0	user		0

5.5.39 Описание регистра otp_mode

Формат регистра otp_mode приведен в следующей таблице:

Таблица 5.45. Поля регистра otp_mode

Диапазон	Название	Описание	Сброс
31:3	-	Резерв	0
2	pd	Отключает линию питания otp, необходимую для ее чтения	1
1	dctrl	Бит разрешает прямое чтение из otp по apb шине. Изменение значения бита должно производится при spi_otp cs==0, или при spi_otp cs==1 и NOP на шине данных	0
0	sp	Режим работы otp с spi_otp 1 - последовательный; 0 - параллельный	0

5.5.40 Описание регистра otp_flag

Формат регистра otp_flag приведен в следующей таблице:

Таблица 5.46. Поля регистра otp_flag

Диапазон	Название	Описание	Сброс
31:2	-	Резерв	0
1	boot_done	Отражает статус завершения процесса инициализации otp памяти при сбросе и валидности регистров fuse0 и fuse1	0 -> 1
0	flag	Отражает состояния вывода FLAG otp	0

5.5.41 Описание регистра fuse0_ecc

Формат регистра fuse0_ecc приведен в следующей таблице:

Таблица 5.47. Поля регистра fuse0_ecc

Диапазон	Название	Описание	Сброс
31:8	-	Резерв	0
7:6	brp	Биты BRP (Bit Repair by Polarity)	0
5:0	ecc	Биты проверки ECC корректности	0

5.5.42 Описание регистра fuse1_ecc

Формат регистра fuse1_ecc приведен в следующей таблице:

Таблица 5.48. Поля регистра fuse1_ecc

Диапазон	Название	Описание	Сброс
31:8	-	Резерв	0
7:6	brp	Биты BRP (Bit Repair by Polarity)	0
5:0	ecc	Биты проверки ECC корректности	0

5.5.43 Описание регистра otp_ecc

Формат регистра otp_ecc приведен в следующей таблице:

Таблица 5.49. Поля регистра otp_ecc

Диапазон	Название	Описание	Сброс
31:8	-	Резерв	0
7:6	brp	Биты BRP (Bit Repair by Polarity)	
5:0	ecc	Биты проверки ECC корректности	0

5.5.44 Описание регистра axicore_internal_priv

Формат регистра axicore_internal_priv приведен в следующей таблице:

Таблица 5.50. Поля регистра axicore_internal_priv

Диапазон	Название	Описание	Сброс
31:5	–	Резерв	?
4:1	privileged	Если в бите поля установлена 1, обращения к слейву с номером бита допустимы только для безопасных транзакций Default 'b1100 Д	0
0	–	Резерв	0

5.5.45 Описание регистра axisys_internal_priv

Формат регистра axisys_internal_priv приведен в следующей таблице:

Таблица 5.51. Поля регистра axisys_internal_priv

Диапазон	Название	Описание	Сброс
31:6	-	Резерв	?
5:1	privileged	Если в бите поля установлена 1, обращения к слейву с номером бита допустимы только для безопасных транзакций Default - 'b11111	0
0	-	Резерв	0

5.5.46 Описание регистра tp_dbgen

Формат регистра tp_dbgen приведен в следующей таблице:

Таблица 5.52. Поля регистра tp_dbgen

Диапазон	Название	Описание	Сброс
31:1	-	Резерв	0
0	risc0_dbgen	Разрешает отладку UltraSoC через RISC0	0

5.5.47 Описание регистра sdr_dbgen

Формат регистра sdr_dbgen приведен в следующей таблице:

Таблица 5.53. Поля регистра sdr_dbgen

Диапазон	Название	Описание	Сброс
31:3	-	Резерв	0
2	dsp1_dbgen	Разрешает отладку UltraSoC DSP1	0
1	dsp0_dbgen	Разрешает отладку UltraSoC DSP0	0
0	risc1_dbgen	Разрешает отладку UltraSoC RISC1	0

5.5.48 Описание регистра sp_dbgen

Формат регистра sp_dbgen приведен в следующей таблице:

Таблица 5.54. Поля регистра sp_dbgen

Диапазон	Название	Описание	Сброс
31:3	-	Резерв	0
2	-	Резерв	?
1	cpu_spiden	Контролирует вход SPIDEN CPU	0
0	cpu_spinden	Контролирует вход SPINDEN CPU	0

5.5.49 Описание регистра s_dbgen

Формат регистра s_dbgen приведен в следующей таблице:

Таблица 5.55. Поля регистра s_dbgen

Диапазон	Название	Описание	Сброс
31:3	-	Резерв	0
2	cpu_suniden	Контролирует вход SUNIDEN CPU	0
1	cpu_dbgen	Контролирует вход DBGEN CPU	0
0	cpu_niden	Контролирует вход NIDEN CPU	0

5.5.50 Описание регистра ust_dbgen

Формат регистра ust_dbgen приведен в следующей таблице:

Таблица 5.56. Поля регистра ust_dbgen

Диапазон	Название	Описание	Сброс
31:8	-	Резерв	0
7	ustc_virt_cons_en	Разрешение работы для компонента VIRT CONS UltraSoC (при vs_en == 1, данный бит игнорируется и работа разрешена)	0
6	ustc_dma_smb_en	Разрешение работы для компонент DMA и SMB UltraSoC (при vs_en == 1, данный бит игнорируется и работа разрешена)	0
5	ustc_usb_jtag_en	Разрешение работы для компонент USB COMM и JTAG COMM UltraSoC (при vs_en == 1, данный бит игнорируется и работа разрешена)	0
4	ustc_acom_en	Разрешение работы для компонента AXI COMM UltraSoC (при vs_en == 1, данный бит игнорируется и работа разрешена)	0
3	ustc_bpam_cpu_en	Разрешение работы для компонента BPAM CPU (при vs_en == 1, данный бит игнорируется и работа разрешена)	0

Диапазон	Название	Описание	Сброс
2	ustc_bpam_risc0_en	Разрешение работы для компонента BPAM RISC0 (при vs_en == 1, данный бит игнорируется и работа разрешена)	0
1	ustc_sdr_en	Разрешение работы для компонент UltraSoC в составе подсистемы sdr (при vs_en == 1, данный бит игнорируется и работа разрешена)	0
0	ustc_ ddr_en	Разрешение работы для компонент UltraSoC в составе подсистемы ddr (при vs_en == 1, данный бит игнорируется и работа разрешена)	0

5.6 Дополнительная информация

Таблица 5.57. Подключение сигналов кросс отладки и GPIO BPAM_SERV

Сигналы BPAM	Подключение
ust_event_i[1:0]	{risc0_debug_o, risc0_de_out_o}
ust_event_out[1:0]	{risc0_break_dsp_i, risc0_de_in_i}
ust_event_out_ack[1:0]	{risc0_en_break_dsp_o, risc0_de_out_o}
ust_gpio[1:0]	axicore_dlock_mst_o[1:0]
ust_gpio[4:2]	axicore_dlock_slv_o[2:0]
ust_gpio[5]	axicore_dlock_wr_o
ust_gpio[6]	axicore_dlock_irq_o
ust_gpio[7]	axicore_dlock_id_o[0]
ust_gpio[15:8]	‘h00
ust_gpio[17:16]	axisys_dlock_mst_o[1:0]
ust_gpio[19:17]	axisys_dlock_slv_o[2:0]
ust_gpio[20]	axisys_dlock_wr_o
ust_gpio[21]	axisys_dlock_irq_o
ust_gpio[24:22]	axisys_dlock_id_o[0]
ust_gpio[31:25]	‘h00
ust_gpio[47:32]	axicore_rresp_s1[1:0]
	axicore_rresp_s2[1:0]
	axicore_rresp_s3[1:0]
	axicore_rresp_s4[1:0]
	axicore_bresp_s1[1:0]
	axicore_bresp_s2[1:0]
	axicore_bresp_s3[1:0]
	axicore_bresp_s4[1:0]
ust_gpio[63:48]	axisys_rresp_s5[1:0]
	axisys_rresp_s2[1:0]
	axisys_rresp_s3[1:0]
	axisys_rresp_s4[1:0]
	axisys_bresp_s5[1:0]
	axisys_bresp_s2[1:0]
	axisys_bresp_s3[1:0]
	axisys_bresp_s4[1:0]

Таблица 5.58. Каналы датчиков температуры и напряжения PVT контроллера

Канал	Подсистема
ts[0], vm[0]	cpu
ts[1], vm[1]	sdr
ts[2], vm[2]	media
ts[3], vm[3]	service

6. ПОДСИСТЕМА SDR

6.1 Общая информация

Подсистемы SDR – это часть микросхемы, которая реализует функцию периферийного радиомодуля перспективных систем связи.

В состав подсистемы входят:

- RISC1 – процессорное ядро подсистемы SDR (описание в Главе 18 «Кластер служебного процессора (RISC)»);
- QLIC1 – контроллер прерываний подсистемы SDR (описание в Главе 19 «Служебный контроллер прерываний (QLIC)»;
- DSP (DSP0, DSP1) – два процессорных ядра Elcore50 (описание в Главе 20 «Нейросетевой акселератор с оптимизацией для сигнатурного анализа и обработки сигналов (DSP)»);
- GNSS – навигационное ядро Navicore5l (описание в Главе 23 «Сопроцессор-ускоритель GNSS»);
- ACC – набор ускорителей (описание в Главе 22 «Ускорители цифровой обработки сигналов»);
    - Витерби декодер (VITDEC),
    - сверточный кодер (CENC),
    - турбо декодер (TDC),
    - турбо кодер (TEC),
    - блок быстрого преобразования Фурье (FFT);
- DFE – блок построения цифровых программно-определяемых приемопередающих трактов с обменом данных по каналам (описание в Главе 21 «Блок цифровой обработки радио (DFE)»):
    - LVDS,
    - CMOS,
    - Jesd204b;

- PCIe (PCIe0, PCIe1) – два контроллера PCI Express 3.0 x4 (описание в Главе 25 «Контроллеры pci-express (PCIE) и jesd204b»);
- Jesd204b – два контроллера Jesd204b (описание в Главе 25 «Контроллеры pci-express (PCIE) и jesd204b»);
- SDRDMA (SDRDMA0, SDRDMA1) – два 8-ми канальных DMA типа память-память (описание в Главе 50 «Контроллер прямого доступа в память подсистемы SDR (SDRDMA)»);
- MPSM – блок многопортовой накристальной общей памяти (4 Мбайта) в виде нескольких банков, а также модуль перемежения/деперемежения данных (INTRLV) по заданному закону (описание в Главе 44 «Блок многопортовой общей памяти, перемежитель (MPSM)»);
- PVT CTR – контроллер накристальных сенсоров температуры и напряжения микросхемы (описание в Главе 48 «Блок накристальных датчиков (sensors)»);
- Компоненты системы отладки:
    - BPAM (BPAM_RISC1, BPAM_DSP0, BPAM_DSP1) – модуль отладки процессора RISC1, DSP0 и DSP1,
    - BUS_MON – модуль контроля системной шины,
    - STATUS_MON – модуль контроля состояния DFE;
- PLL подсистемы;
- URB – блок управляющих регистров подсистемы;
- UCG – блок формирования тактовых частот подсистемы (описание в Главе 43 «Универсальный блок формирования тактовой частоты (UCG)»);
- MAILBOX1 – почтовый ящик для обращения к контуру SDR (описание в Главе 41 «Блок обмена сообщениями (mailbox)»);
- SPINLOCK – блок поддержки атомарных операций (описание в Главе 42 «Блок атомарных операций (spinlock)»);
- TIMERS1 – набор системных таймеров (описание в Главе 39 «Блок универсальных таймеров (timers)»);
- Локальный коммутатор в виде набора компонент.

6.1.1 Структурная схема

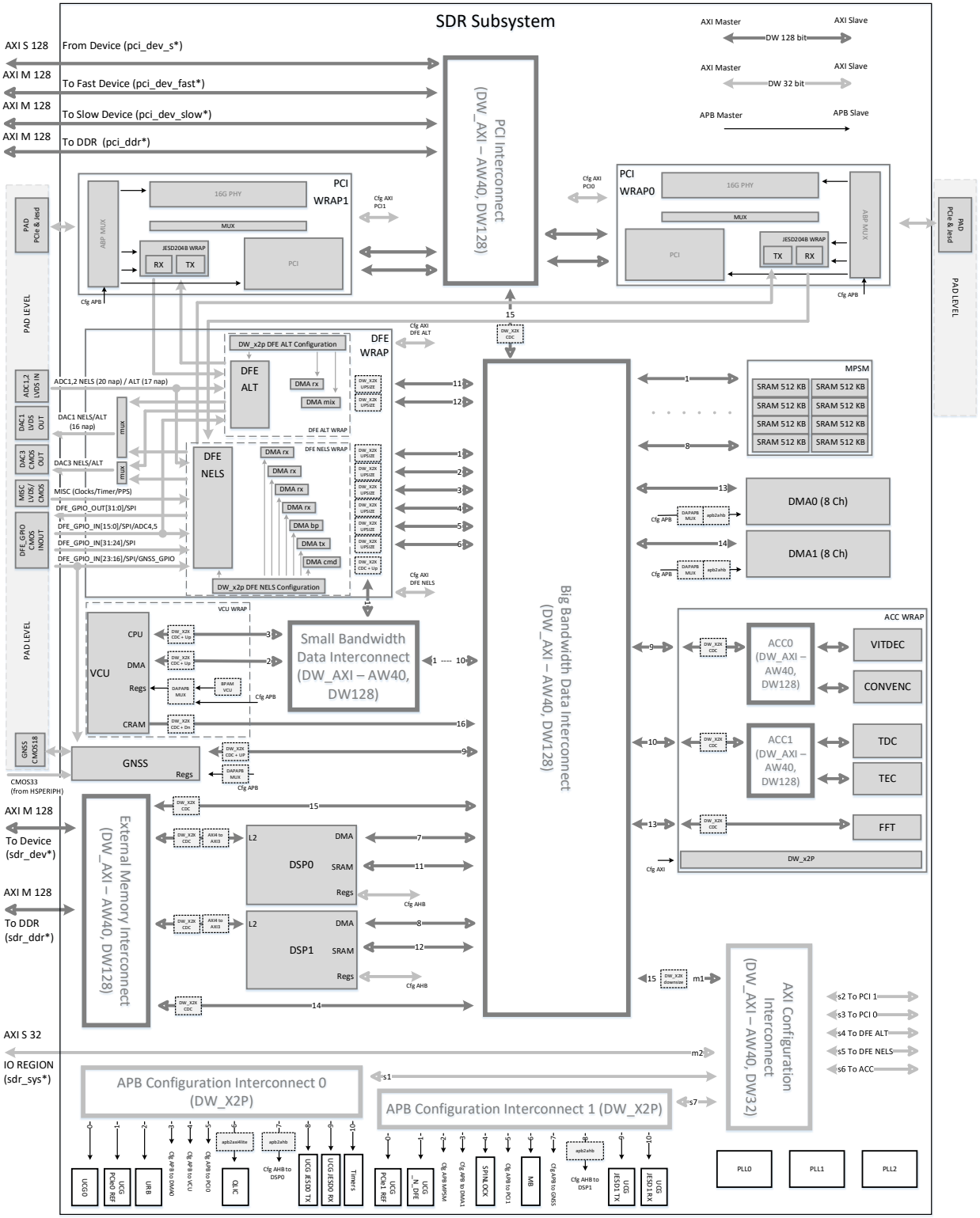


Рисунок 6.1. Структурная схема

6.1.2 Домены синхронизации

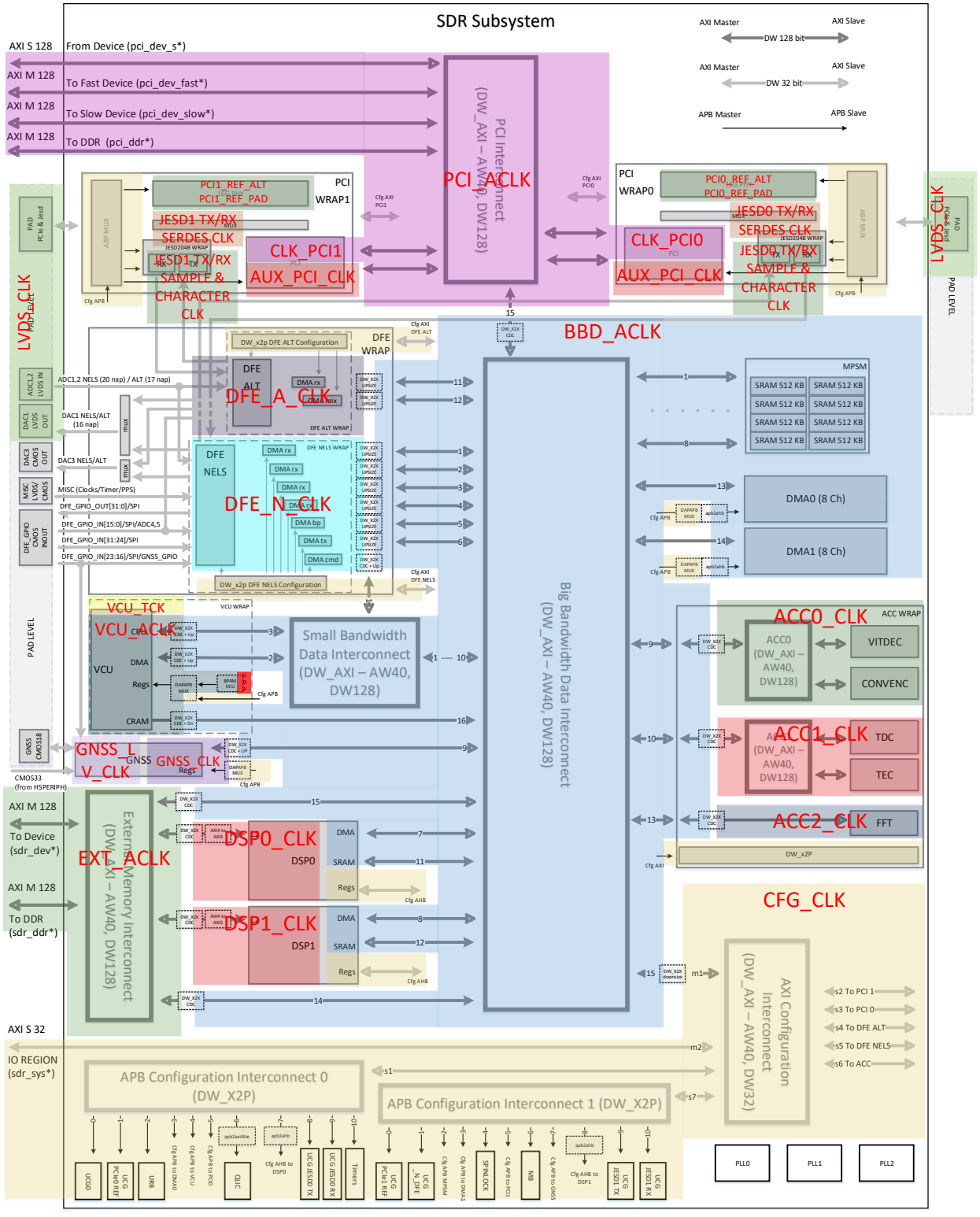


Рисунок 6.2. Домены синхронизации

Структурная схема системы синхронизации и сброса (часть 1) представлена на Рисунок 6.3.

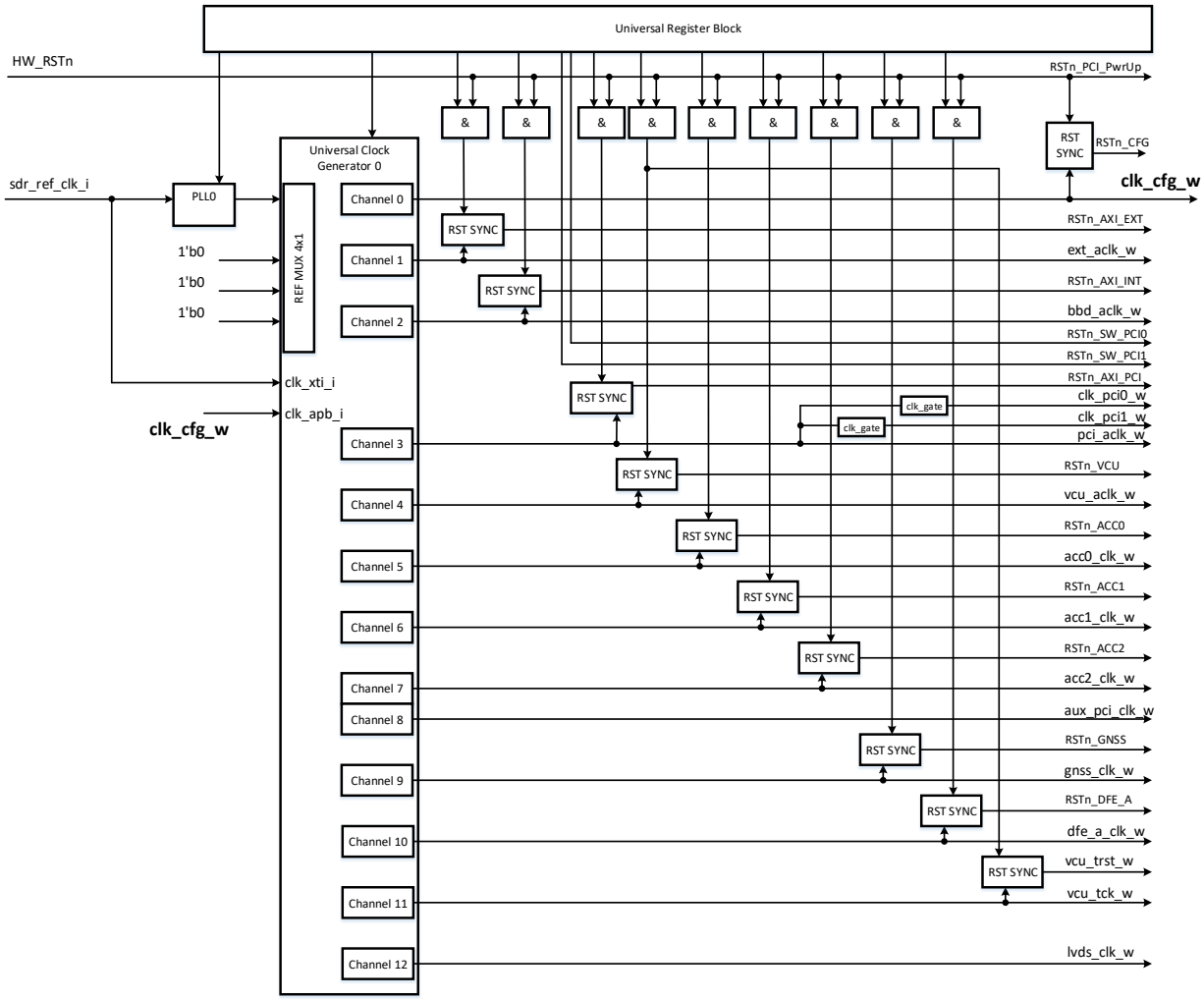


Рисунок 6.3. Схема системы синхронизации (часть 1)

Структурная схема системы синхронизации и сброса (часть 2) представлена на Рисунок 6.4.

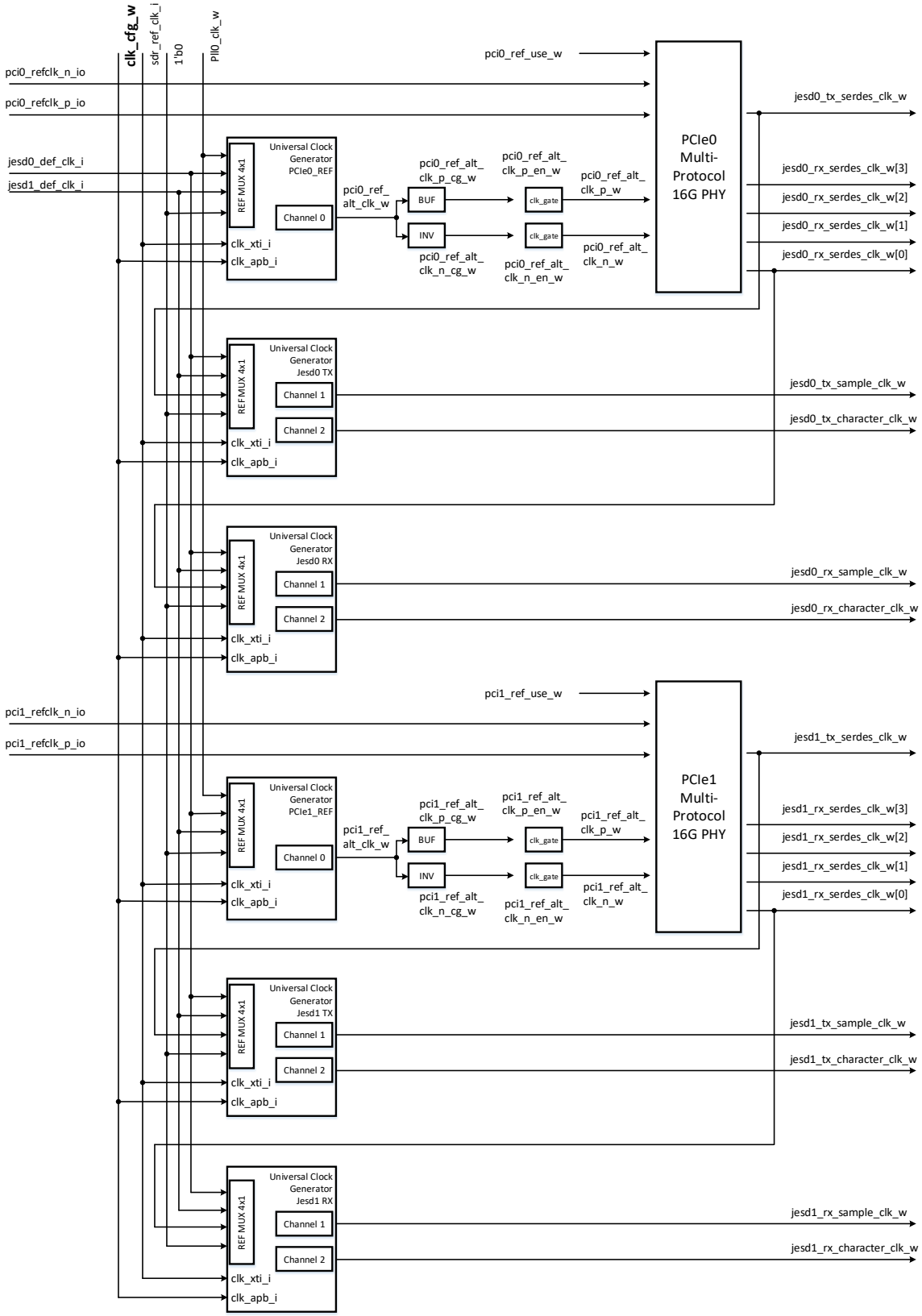


Рисунок 6.4. Схема системы синхронизации (часть 2)

Структурная схема системы синхронизации и сброса (часть 3) представлена на Рисунок 6.5

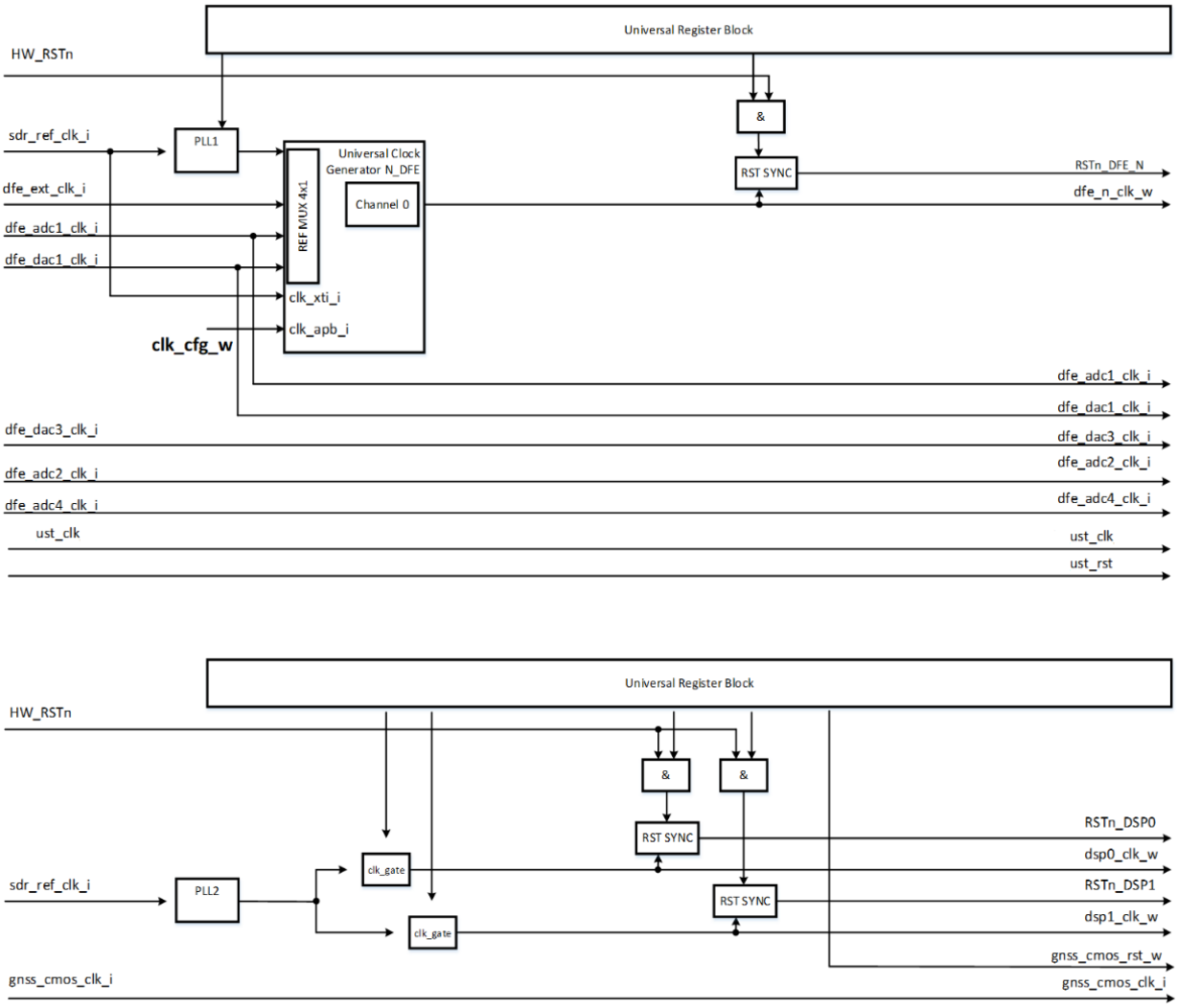


Рисунок 6.5. Схема системы синхронизации (часть 3)

6.2 Карта памяти подсистемы SDR

0x0190_0000 - 0x027F_FFFF (15 Мбайт) – конфигурационные регистры

0x0300_0000 - 0x03FF_FFFF (16 Мбайт) – накристальная память

Таблица 6.1. Карта памяти подсистемы SDR

Область конфигурационных регистров			
Начало	Конец	Объем	Описание
0x0190_0000	0x0190_3FFF	16KB	UCG0
0x0190_4000	0x0190_7FFF	16KB	-
0x0190_8000	0x0190_BFFF	16KB	UCG_PCIe0_REF
0x0190_C000	0x0190_FFFF	16KB	-
0x0191_0000	0x0191_3FFF	16KB	URB
0x0191_4000	0x0191_7FFF	16KB	-
0x0191_8000	0x0191_BFFF	16KB	SDRDMA0
0x0191_C000	0x0191_FFFF	16KB	-
0x0192_0000	0x0192_3FFF	16KB	RISC1 QDMA
0x0192_4000	0x0192_7FFF	16KB	RISC1 VMMU
0x0192_8000	0x0192_BFFF	16KB	RISC1 CSR
0x0192_C000	0x0192_FFFF	16KB	RISC1 OnCD
0x0193_0000	0x0193_FFFF	32KB	-
0x0193_8000	0x0193_83FF	1KB	TIMERS1
0x0193_8400	0x0193_FFFF	31KB	-
0x0194_0000	0x0197_FFFF	256KB	QLIC1
0x0198_0000	0x019B_FFFF	256KB	DSP0
0x019C_0000	0x019D_FFFF	128KB	-
0x019E_0000	0x019E_3FFF	16KB	UCG_JESD0_TX
0x019E_4000	0x019E_7FFF	16KB	-
0x019E_8000	0x019E_BFFF	16KB	UCG_JESD0_RX
0x019E_C000	0x019F_F7FF	78KB	-
0x019F_F800	0x019F_FBFF	1KB	JESD0_RX_CFG
0x019F_FC00	0x019F_FFFF	1KB	JESD0_TX_CFG
0x01A0_0000	0x01A0_07FF	2KB	PCI0_APB_CTR
0x01A0_0800	0x01AF_FFFF	1MB - 2KB	-
0x01B0_0000	0x01B0_0FFF	4KB	Alt_dma_rx
0x01B0_1000	0x01B0_1FFF	4KB	Alt_dma_mix
0x01B0_2000	0x01B1_FFFF	120KB	-
0x01B2_0000	0x01B3_FFFF	128KB	Alt_dfe
0x01B4_0000	0x01B7_FFFF	256KB	-

Область конфигурационных регистров			
Начало	Конец	Объем	Описание
0x01B8_0000	0x01B8_0FFF	4KB	Nels_dma_ddc0
0x01B8_1000	0x01B8_1FFF	4KB	Nels_dma_ddc1
0x01B8_2000	0x01B8_2FFF	4KB	Nels_dma_ddc2
0x01B8_3000	0x01B8_3FFF	4KB	Nels_dma_ddc3
0x01B8_4000	0x01B8_4FFF	4KB	Nels_dma_adcbp
0x01B8_5000	0x01B8_5FFF	4KB	Nels_dma_duc
0x01B8_6000	0x01B8_6FFF	4KB	Nels_dma_cmd
0x01B8_7000	0x01B8_7FFF	4KB	Nels_dfe
0x01B8_8000	0x01BF_FFFF	512KB - 32KB	-
0x01C0_0000	0x01C0_3FFF	16KB	UCG_PCIe1_REF
0x01C0_4000	0x01C0_7FFF	16KB	-
0x01C0_8000	0x01C0_BFFF	16KB	UCG_N_DFE
0x01C0_C000	0x01C0_FFFF	16KB	-
0x01C1_0000	0x01C1_3FFF	16KB	MPSM
0x01C1_4000	0x01C1_7FFF	16KB	-
0x01C1_8000	0x01C1_BFFF	16KB	SDRDMA1
0x01C1_C000	0x01C1_FFFF	16KB	-
0x01C2_0000	0x01C2_FFFF	64KB	Spinlock
0x01C3_0000	0x01C5_FFFF	192KB	-
0x01C6_0000	0x01C6_FFFF	64KB	GNSS
0x01C7_0000	0x01C7_FFFF	64KB	-
0x01C8_0000	0x01CB_FFFF	256KB	DSP1
0x01CC_0000	0x01CD_FFFF	128KB	-
0x01CE_0000	0x01CE_3FFF	16KB	UCG_JESD1_TX
0x01CE_4000	0x01CE_7FFF	16KB	-
0x01CE_8000	0x01CE_BFFF	16KB	UCG_JESD1_RX
0x01CC_0000	0x01CF_FFFF	78KB	-
0x01CF_F800	0x01CF_FBFF	1KB	JESD1_RX_CFG
0x01CF_FC00	0x01CF_FFFF	1KB	JESD1_TX_CFG
0x01D0_0000	0x01D0_07FF	2KB	PCI1_APB_CTR
0x01D0_0800	0x01DF_FFFF	1MB - 2KB	-
0x01E0_0000	0x01E1_FFFF	128KB	Mailbox1
0x01E2_0000	0x01EF_FFFF	1MB - 128KB	-
0x01F0_0000	0x01F0_03FF	1KB	VITDEC
0x01F0_0400	0x01F0_07FF	1KB	CONVENC
0x01F0_0800	0x01F0_0BFF	1KB	TDC
0x01F0_0C00	0x01F0_0FFF	1KB	TEC
0x01F0_1000	0x01F0_13FF	1KB	FFT

Область конфигурационных регистров			
Начало	Конец	Объем	Описание
0x01F0_1400	0x01FF_FFFF	1MB – 5KB	-
0x0200_0000	0x023F_FFFF	4MB	PCI0_DBI
0x0240_0000	0x027F_FFFF	4MB	PCI1_DBI
0x0280_0000	0x0283_FFFF	256KB	Резерв DSP_LOCAL_REGS
Область накристальной памяти			
0x02E0_0000	0x02FF_FFFF	2MB	Резерв DSP_LOCAL_XYRAM + PRAM
0x0300_0000	0x031F_FFFF	2MB	DSP0_XYRAM + PRAM
0x0320_0000	0x033F_FFFF	2MB	DSP1_XYRAM + PRAM
0x0340_0000	0x037F_FFFF	4MB	MPSM_SRAM
0x0380_0000	0x0380_0FFF	4KB	VITDEC
0x0380_1000	0x0380_1FFF	4KB	CONVENC
0x0380_2000	0x038F_FFFF	1MB - 8KB	-
0x0390_0000	0x0393_FFFF	256KB	TDC
0x0394_0000	0x0394_FFFF	64KB	TEC
0x0398_0000	0x039F_FFFF	704KB	-
0x03A0_0000	0x03A0_FFFFf	64KB	FFT
0x03A0_1000	0x03AF_FFFF	1MB - 64KB	-
0x03B0_0000	0x03B0_7FFF	32KB	CRAM*
0x03B0_8000	0x03FF_FFFF	5MB – 32KB	-
PCI			
0x00_6000_0000	0x00_6FFF_FFFF	256MB	PCIe0 Low
0x00_7000_0000	0x00_7FFF_FFFF	256MB	PCIe1 Low
0x04_0000_0000	0x05_BFFF_FFFF	7GB	PCIe0 Mid
0x05_C000_0000	0x07_7FFF_FFFF	7GB	PCIe1 Mid
0x80_0000_0000	0xBF_FFFF_FFFF	256GB	PCIe0 High
0xC0_0000_0000	0xFF_FFFF_FFFF	256GB	PCIe1 High

*Байтовые обращения к памяти CRAM не поддерживаются

6.3 Регистры URB

Перечень регистров URB приведен в Таблица 6.2.

Таблица 6.2. Перечень программно-доступных регистров URB

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
sdr_pll0_ctl	Регистр конфигурации PLL0	0x0	RW	0x0000
sdr_pll0_diag	Регистр диагностики PLL0	0x2	RW	0x0004
sdr_pll1_ctl	Регистр конфигурации PLL1	0x0	RW	0x0008
sdr_pll1_diag	Регистр диагностики PLL1	0x2	RW	0x000C
sdr_pll2_ctl	Регистр конфигурации pll2	0x0	RW	0x0010
sdr_pll2_diag	Регистр диагностики pll2	0x2	RW	0x0014
sdr_ucg_pci0_ref_ctl	UCG PCI0 REF Control Register	0x1	RW	0x0020
sdr_ucg_jesd0_tx_ref_ctl	UCG JESD0_TX REF Control Register	0x0	RW	0x0024
sdr_ucg_jesd0_rx_ref_ctl	UCG JESD0_RX REF Control Register	0x1	RW	0x0028
sdr_jesd0_ctl	Регистр управления Jesd0	0x2	RW	0x002C
sdr_ucg_pci1_ref_ctl	UCG PCI1 REF Control Register	0x1	RW	0x0030
sdr_ucg_jesd1_tx_ref_ctl	UCG JESD1_TX REF Control Register	0x0	RW	0x0034
sdr_ucg_jesd1_rx_ref_ctl	UCG JESD1_RX REF Control Register	0x1	RW	0x0038
sdr_jesd1_ctl	Регистр управления Jesd1	0x2	RW	0x003C
sdr_ucg_n_dfe_ctl	UCG_N_DFE Control Register	0x0	RW	0x0040
sdr_n_dfe_ctl	N DFE Control Register	0x0	RW	0x0044
sdr_csr_dsp	CSR_DSP Register	0x0	RW	0x0048
sdr_dsp_ctl	DSP Control Register	0x0	RW	0x004C
sdr_pci0_ctl	PCI0 Control Register	0x92	RW	0x0050
sdr_pci1_ctl	PCI1 Control Register	0x92	RW	0x0054
sdr_dma0_ctl	DMA0 Control Register	0x1020	RW	0x0058
sdr_dma1_ctl	DMA1 Control Register	0x1020	RW	0x005C
sdr_dsp0_ppolicy	Регистр установки состояния ядра 0 DSP	0x0	RW	0x0060
sdr_dsp0_pstatus	Регистр статуса состояния ядра 0 DSP	0x0	RO	0x0064
sdr_dsp1_ppolicy	Регистр установки состояния ядра 1 DSP	0x0	RW	0x0068
sdr_dsp1_pstatus	Регистр статуса состояния ядра 1 DSP	0x0	RO	0x006C
sdr_risc1_ppolicy	Регистр установки состояния ядра RISC1	0x0	RW	0x0070
sdr_risc1_pstatus	Регистр статуса состояния ядра RISC1	0x0	RO	0x0074
sdr_acc_rst_ctl	ACC SW_RST Register	0x0	RW	0x0078
sdr_gnss_rst_ctl	GNSS SW_RST Register	0x3	RW	0x007C
sdr_a_dfe_rst_ctl	A DFE SW_RST Register	0x0	RW	0x0080
sdr_jesd0_rst_ctl	JESD0 SW_RST Register	0x0	RW	0x0084
sdr_jesd1_rst_ctl	JESD1 SW_RST Register	0x0	RW	0x0088
sdr_n_dfe_rst_ctl	N DFE SW_RST Register	0x0	RW	0x008C
sdr_ext_rst_ctl	External Interconnect SW_RST Register	0x0	RW	0x0090
sdr_bbd_rst_ctl	Big Bandwidth Data Interconnect SW_RST Register	0x0	RW	0x0094
sdr_pci_rst_ctl	PCI Interconnect SW_RST Register	0x0	RW	0x0098
SDR_TRUST_ZONE	Регистр управления доверенной зоной SDR	0x0	RW	0x00A0
DFE_TRUST_ZONE	Регистр управления доверенной зоной DFE	0x0	RW	0x00A4

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
LVDS_SEG1_CTL	Регистр управления контактными площадками RTT сегмента 1	0x20001	RW	0x0110
LVDS_SEG1_STATUS	Регистр статуса контактных площадок RTT сегмента 1	0x0	RO	0x0114
LVDS_SEG1_RX_EN_CTL	Регистр управления контактными площадками RX сегмента 1 поля EN	0x0	RW	0x0118
LVDS_SEG1_RX_HYST_EN_CTL	Регистр управления контактными площадками RX сегмента 1 поля HYST_EN	0x0	RW	0x011C
LVDS_SEG2_CTL	Регистр управления контактными площадками REF сегмента 2	0x20000	RW	0x0120
LVDS_SEG2_TX_OE_CTL	Регистр управления контактными площадками TX сегмента 2 поля EN	0x0	RW	0x0124
LVDS_SEG3_CTL	Регистр управления контактными площадками RTT сегмента 3	0x20001	RW	0x0128
LVDS_SEG3_STATUS	Регистр статуса контактных площадок RTT сегмента 3	0x0	RO	0x012C
LVDS_SEG3_RX_EN_CTL	Регистр управления контактными площадками RX сегмента 3 поля EN	0x0	RW	0x0130
LVDS_SEG3_RX_HYST_EN_CTL	Регистр управления контактными площадками RX сегмента 3 поля HYST_EN	0x0	RW	0x0134
CMOS_GPIO_REN_CTL	CMOS GPIO Pull-resistor enable	0xffff_ffff	RW	0x0138
CMOS_MIX_REN_CTL	CMOS MIX Pull-resistor enable	0xff_ffff	RW	0x013C
SOFT_NMI_SET	Регистр установки прерывания NMI RISC1	0x0	RW	0x0140
SOFT_NMI_CLEAR	Регистр снятия прерывания NMI RISC1	0x0	RW	0x0144
SOFT_NMI_MASK	Регистр маски прерывания NMI RISC1	0x1	RW	0x0148
SOFT_NMI_STATUS	Регистр статуса прерывания NMI RISC1	0x0	RO	0x014C
PCIE_PERSTn_CTL	Регистр управления сбросом PERSTn	0x101	RW	0x0150

6.3.1 Регистр sdr_pll0_ctl

Регистр sdr_pll0_ctl.

Таблица 6.3. Формат регистра sdr_pll0_ctl

Номер бита	Условное обозначение	Назначение
31	pll_lock	Бит указывает окончание настройки PLL на новую заданную частоту работы. При начале процедуры настройки либо при выключенной PLL сбрасывается в 0.
30:27	nr_man	Коэффициент nr настройки pll
26:14	nf_man	Коэффициент nf настройки pll
13:10	od_man	Коэффициент od настройки pll
9	man	При бите заданном в 1 и sel > 0 конфигурация pll определяется полям *_man данного регистра. Выходная частота определяется по формуле ref_clk * (nf_man + 1) / (nr_man + 1) / (od_man + 1). Поля *_man рекомендуется задавать исходя из значений генерируемым скриптом настройки pll. Значение выходной частоты не должно превышать 3600 МГц.
8	-	Резерв

Номер бита	Условное обозначение	Назначение
7:0	pll_sel	Определяет выходную частоту и режим работы PLL: • 0x0 - опорная частота ref_clk, PLL выключена; • 0x1 - ref_clk * 2; • 0x2 - ref_clk * 3; • 0x3 - ref_clk * 4; • ... - ...; • 0x73 - ref_clk * 116; • 0x74 - ref_clk * 116; • ... - ...; • 0xFF - ref_clk * 116

6.3.2 Регистр sdr_pll0_diag

Регистр sdr_pll0_diag.

Таблица 6.4. Формат регистра sdr_pll0_diag

Номер бита	Условное обозначение	Назначение
31:5	-	Резерв
4	fbslip	Диагностический выход pll - регистрирует ситуацию feedback cycle slip
3	rfslip	Диагностический выход pll - регистрирует ситуацию referency cycle slip
2	fasten	Диагностический вход pll - разрешает механизм fast lock
1	ensat	Диагностический вход pll - разрешает насыщение, в нормальном режиме - 1
0	test	Диагностический вход pll - bypass опорной частоты pll на выход при 1

6.3.3 Регистр sdr_pll1_ctl

Регистр sdr_pll1_ctl.

Таблица 6.5. Формат регистра sdr_pll1_ctl

Номер бита	Условное обозначение	Назначение
31	pll_lock	Бит указывает окончание настройки PLL на новую заданную частоту работы. При начале процедуры настройки либо при выключенной PLL сбрасывается в 0.
30:27	nr_man	Коэффициент nr настройки pll
26:14	nf_man	Коэффициент nf настройки pll
13:10	od_man	Коэффициент od настройки pll
9	man	При бите заданном в 1 и sel > 0 конфигурация pll определяется полям *_man данного регистра. Выходная частота определяется по формуле ref_clk * (nf_man + 1) / (nr_man + 1) / (od_man + 1). Поля *_man рекомендуется задавать исходя из значений генерируемым скриптом настройки pll. Значение выходной частоты не должно превышать 3600 МГц.
8	-	Резерв

Номер бита	Условное обозначение	Назначение
7:0	pll_sel	Определяет выходную частоту и режим работы PLL: • 0x0 - опорная частота ref_clk, PLL выключена; • 0x1 - ref_clk * 2; • 0x2 - ref_clk * 3; • 0x3 - ref_clk * 4; • ... - ...; • 0x73 - ref_clk * 116; • 0x74 - ref_clk * 116; • ... - ...; • 0xFF - ref_clk * 116

6.3.4 Регистр sdr_pll1_diag

Регистр sdr_pll1_diag.

Таблица 6.6. Формат регистра sdr_pll1_diag

Номер бита	Условное обозначение	Назначение
31:5	-	Резерв
4	fbslip	Диагностический выход pll - регистрирует ситуацию feedback cycle slip
3	rfslip	Диагностический выход pll - регистрирует ситуацию referency cycle slip
2	fasten	Диагностический вход pll - разрешает механизм fast lock
1	ensat	Диагностический вход pll - разрешает насыщение, в нормальном режиме - 1
0	test	Диагностический вход pll - bypass опорной частоты pll на выход при 1

6.3.5 Регистр sdr_pll2_ctl

Регистр sdr_pll2_ctl.

Таблица 6.7. Формат регистра sdr_pll2_ctl

Номер бита	Условное обозначение	Назначение
31	pll_lock	Бит указывает окончание настройки PLL на новую заданную частоту работы. При начале процедуры настройки либо при выключенной PLL сбрасывается в 0.
30:27	nr_man	Коэффициент nr настройки pll
26:14	nf_man	Коэффициент nf настройки pll
13:10	od_man	Коэффициент od настройки pll
9	man	При бите заданном в 1 и sel > 0 конфигурация pll определяется полям *_man данного регистра. Выходная частота определяется по формуле ref_clk * (nf_man + 1) / (nr_man + 1) / (od_man + 1). Поля *_man рекомендуется задавать исходя из значений генерируемым скриптом настройки pll. Значение выходной частоты не должно превышать 3600 МГц.
8	-	Резерв

Номер бита	Условное обозначение	Назначение
7:0	pll_sel	Определяет выходную частоту и режим работы PLL: • 0x0 - опорная частота ref_clk, PLL выключена; • 0x1 - ref_clk * 2; • 0x2 - ref_clk * 3; • 0x3 - ref_clk * 4; • ... - ...; • 0x73 - ref_clk * 116; • 0x74 - ref_clk * 116; • ... - ...; • 0xFF - ref_clk * 116

6.3.6 Регистр sdr_pll2_diag

Регистр sdr_pll2_diag.

Таблица 6.8. Формат регистра sdr_pll2_diag

Номер бита	Условное обозначение	Назначение
31:5	-	Резерв
4	fbslip	Диагностический выход pll - регистрирует ситуацию feedback cycle slip
3	rfslip	Диагностический выход pll - регистрирует ситуацию referency cycle slip
2	fasten	Диагностический вход pll - разрешает механизм fast lock
1	ensat	Диагностический вход pll - разрешает насыщение, в нормальном режиме - 1
0	test	Диагностический вход pll - bypass опорной частоты pll на выход при 1

6.3.7 Регистр sdr_ucg_pci0_ref_ctl

Регистр sdr_ucg_pci0_ref_ctl.

Таблица 6.9. Формат регистра sdr_ucg_pci0_ref_ctl

Номер бита	Условное обозначение	Назначение
31:9	-	Резерв
8	ref_rst_n_sw	Глобальный сброс блока sdr_ucg_pci0_ref_ctl. Активный уровень 0.
7:2	-	Резерв
1:0	sel_clk_ref	Выбор референсной частоты: 2'b00 - pll0_clk 2'b01 – jesd0_dev_clk 2'b10 - jesd1_dev_clk

6.3.8 Регистр sdr_ucg_jesd0_tx_ref_ctl

Регистр sdr_ucg_jesd0_tx_ref_ctl.

Таблица 6.10. Формат регистра sdr_ucg_jesd0_tx_ref_ctl

Номер бита	Условное обозначение	Назначение
31:9	-	Резерв

Номер бита	Условное обозначение	Назначение
8	ref_rst_n_sw	Глобальный сброс блока sdr_ucg_jesd0_tx_ref_ctl. Активный уровень 0.
7:2	-	Резерв
1:0	sel_clk_ref	Выбор референсной частоты: 2'b00 - jesd0_dev_clk 2'b01 - jesd1_dev_clk 2'b10 - jesd0_tx_serdes_clk

6.3.9 Регистр sdr_ucg_jesd0_rx_ref_ctl

Регистр sdr_ucg_jesd0_rx_ref_ctl.

Таблица 6.11. Формат регистра sdr_ucg_jesd0_rx_ref_ctl

Номер бита	Условное обозначение	Назначение
31:9	-	Резерв
8	ref_rst_n_sw	Глобальный сброс блока sdr_ucg_jesd0_rx_ref_ctl. Активный уровень 0.
7:2	-	Резерв
1:0	sel_clk_ref	Выбор референсной частоты: 2'b00 - jesd0_dev_clk 2'b01 - jesd1_dev_clk 2'b10 - jesd0_rx_serdes_clk

6.3.10 Регистр sdr_jesd0_ctl

Регистр sdr_jesd0_ctl.

Таблица 6.12. Формат регистра sdr_jesd0_ctl

Номер бита	Условное обозначение	Назначение
31:2	-	Резерв
1	sysref_rx_sel	Выбор источника сигнала SYSREF для RX: 1'b0 - КП JESD0_SYSREF 1'b1 - КП JESD1_SYSREF
0	sysref_tx_sel	Выбор источника сигнала SYSREF для TX: 1'b0 - КП JESD0_SYSREF 1'b1 - КП JESD1_SYSREF

6.3.11 Регистр sdr_ucg_pci1_ref_ctl

Регистр sdr_ucg_pci1_ref_ctl.

Таблица 6.13. Формат регистра sdr_ucg_pci1_ref_ctl

Номер бита	Условное обозначение	Назначение
31:9	-	Резерв
8	ref_rst_n_sw	Глобальный сброс блока sdr_ucg_pci1_ref_ctl. Активный уровень 0.
7:2	-	Резерв

Номер бита	Условное обозначение	Назначение
1:0	sel_clk_ref	Выбор референсной частоты: 2'b00 - pll0_clk 2'b01 – jesd0_dev_clk 2'b10 - jesd1_dev_clk

6.3.12 Регистр sdr_ucg_jesd1_tx_ref_ctl

Регистр sdr_ucg_jesd1_tx_ref_ctl.

Таблица 6.14. Формат регистра sdr_ucg_jesd1_tx_ref_ctl

Номер бита	Условное обозначение	Назначение
31:9	-	Резерв
8	ref_rst_n_sw	Глобальный сброс блока sdr_ucg_jesd1_tx_ref_ctl. Активный уровень 0.
7:2	-	Резерв
1:0	sel_clk_ref	Выбор референсной частоты: 2'b00 - jesd0_dev_clk 2'b01 - jesd1_dev_clk 2'b10 – jesd1_tx_serdes_clk

6.3.13 Регистр sdr_ucg_jesd1_rx_ref_ctl

Регистр sdr_ucg_jesd1_rx_ref_ctl.

Таблица 6.15. Формат регистра sdr_ucg_jesd1_rx_ref_ctl

Номер бита	Условное обозначение	Назначение
31:9	-	Резерв
8	ref_rst_n_sw	Глобальный сброс блока sdr_ucg_jesd1_rx_ref_ctl. Активный уровень 0.
7:2	-	Резерв
1:0	sel_clk_ref	Выбор референсной частоты: 2'b00 - jesd0_dev_clk 2'b01 - jesd1_dev_clk 2'b10 – jesd1_rx_serdes_clk

6.3.14 Регистр sdr_jesd1_ctl

Регистр sdr_jesd1_ctl.

Таблица 6.16. Формат регистра sdr_jesd1_ctl

Номер бита	Условное обозначение	Назначение
31:2	-	Резерв
1	sysref_rx_sel	Выбор источника сигнала SYSREF для RX: 1'b0 - КП JESD0_SYSREF 1'b1 - КП JESD1_SYSREF
0	sysref_tx_sel	Выбор источника сигнала SYSREF для TX: 1'b0 - КП JESD0_SYSREF 1'b1 - КП JESD1_SYSREF

6.3.15 Регистр sdr_ucg_n_dfe_ctl

Регистр sdr_ucg_n_dfe_ctl.

Таблица 6.17. Формат регистра sdr_ucg_n_dfe_ctl

Номер бита	Условное обозначение	Назначение
31:9	-	Резерв
8	ref_rst_n_sw	Глобальный сброс блока sdr_ucg_n_dfe_ctl. Активный уровень 0.
7:2	-	Резерв
1:0	sel_clk_ref	Выбор референсной частоты: 2'b00 – pll1_clk 2'b01 - dfe_ext_clk 2'b10 - dfe_adc1_clk 2'b11 - dfe_dac1_clk

6.3.16 Регистр sdr_n_dfe_ctl

Регистр sdr_n_dfe_ctl.

Таблица 6.18. Формат регистра sdr_n_dfe_ctl

Номер бита	Условное обозначение	Назначение
31	dfe_alt_dac_cmos_en	Выбор подключения к порту DFE_DAC3 1 - DFE_ALT 0 - DFE_NELS
30	dfe_alt_dac_lvds_en	Выбор подключения к порту DFE_DAC1 1 - DFE_ALT 0 - DFE_NELS
29:2	-	Резерв
1:0	gpio_mode	Выбор режима работы GPIO: 00 - GPIO 01 - SSI 10 - SSI + ADC 11 - GPIO + ADC

6.3.17 Регистр sdr_csr_dsp

Регистр sdr_csr_dsp.

Таблица 6.19. Формат регистра sdr_csr_dsp

Номер бита	Условное обозначение	Назначение
31:10	-	Резерв
9:8	MSK	Маска MSK[0] - маска DSP0 MSK[1] - маска DSP1
7:5	-	Резерв

Номер бита	Условное обозначение	Назначение
4:1	ACT	Действие: 4'b0000 - запуск (DCSR[i].RUN = 1) 4'b0001 - останов (DCSR[i].RUN = 0) 4'b0010 - запуск (режим отладки) (dbDCSR[i].dbRUN = 1) 4'b0011 - останов (режим отладки) (dbDCSR[i].dbRUN = 0) 4'b0100 - перевод в режим отладки (dbDCSR[i].RDBG = 1) 4'b0101 - выход из режима отладки (dbDCSR[i].RDBG = 0) 4'b0110 - разрешение прерываний (DCSR[i].IE = 1, dbDCSR[i].dbIE = 1) 4'b0111 - запрет прерываний (DCSR[i].IE = 0, dbDCSR[i].dbIE = 0) 4'b1000 - останов с запретом прерываний (DCSR[i].RUN = 0, DCSR[i].IE = 0, dbDCSR[i].dbIE = 0) 4'b1001 - переход в отладчик (dbDCSR[i].dbBRK = 1) 4'b1010 - запустить синхронную пошаговую отладку (регистры dbCNTR предварительно установлены внешним отладчиком) (dbDCSR[i].RDBG = 1, dbDCSR[i].dbRUN = 1) 4'b1011 - 4'b1111 - нет действий
0	SYN	При записи в поле SYN значения «1» будет произведено действие над всеми ядрами кластера согласно полю ACT. При этом проверяется маска для каждого ядра DSP[i] и действие исполняется только при условии, что маска была установлена(MSK[i] = «1») Значение поля SYN сбрасывается в «0» автомитически после записи «1»

6.3.18 Регистр sdr_dsp_ctl

Регистр sdr_dsp_ctl.

Таблица 6.20. Формат регистра sdr_dsp_ctl

Номер бита	Условное обозначение	Назначение
31:10	-	Резерв
9	dsp1_clk_en	Управление синхросигналом для DSP0 «1» - частота включена «0» - частота выключена
8	dsp0_clk_en	Управление синхросигналом для DSP0 «1» - частота включена «0» - частота выключена
7:0	-	Резерв

6.3.19 Регистр sdr_pci0_ctl

Регистр sdr_pci0_ctl.

Таблица 6.21. Формат регистра sdr_pci0_ctl

Номер бита	Условное обозначение	Назначение
31:10	-	Резерв
9	dbi_csysack	Low Power Interface
8	dbi_cactive	Low Power Interface
7	dbi_csysreq	Low Power Interface
6:5	-	Резерв

Номер бита	Условное обозначение	Назначение
4	ref_use_pad	Выбор источника референсной частоты. 0 - ref_alt; 1 - ref_pad
3	ref_alt_clk_n_en	Включение чатоты ref_alt_clk_n_en: 0 - частота выключена 1 - частота включена Изменение значения данного поля корректно осуществлять после остановки частоты ref_alt_clk (см. описание UCG3)
2	ref_alt_clk_p_en	Включение чатоты ref_alt_clk_p_en: 0 - частота выключена 1 - частота включена Изменение значения данного поля корректно осуществлять после остановки частоты ref_alt_clk (см. описание UCG3)
1	button_rst_n	Управление сигналом сброса button_rst_n
0	clk_en	Включение чатоты PCI: 0 - частота выключена 1 - частота включена

6.3.20 Регистр sdr_pci1_ctl

Регистр sdr_pci1_ctl.

Таблица 6.22. Формат регистра sdr_pci1_ctl

Номер бита	Условное обозначение	Назначение
31:10	-	Резерв
9	dbi_csysack	Low Power Interface
8	dbi_cactive	Low Power Interface
7	dbi_csysreq	Low Power Interface
6:5	-	Резерв
4	ref_use_pad	Выбор источника референсной частоты. 0 - ref_alt; 1 - ref_pad
3	ref_alt_clk_n_en	Включение чатоты ref_alt_clk_n_en: 0 - частота выключена 1 - частота включена Изменение значения данного поля корректно осуществлять после остановки частоты ref_alt_clk (см. описание UCG1)
2	ref_alt_clk_p_en	Включение чатоты ref_alt_clk_p_en: 0 - частота выключена 1 - частота включена Изменение значения данного поля корректно осуществлять после остановки частоты ref_alt_clk (см. описание UCG1)
1	button_rst_n	Управление сигналом сброса button_rst_n
0	clk_en	Включение чатоты PCI: 0 - частота выключена 1 - частота включена

6.3.21 Регистр sdr_dma0_ctl

Регистр sdr_dma0_ctl.

Таблица 6.23. Формат регистра sdr_dma0_ctl

Номер бита	Условное обозначение	Назначение
31:24	dmar_status	Статус запросов DMA Сброс осуществляется записью "1". После записи автоматически сбрасывается в "0" dmar_status[0] - статус запроса 0-го канала DMA dmar_status[1] - статус запроса 1-го канала DMA и т.д.
23:16	dmar_clear	Сброс запросов DMA Сброс осуществляется записью "1". После записи автоматически сбрасывается в "0" dmar_clear[0] - сбрасывает запрос 0-го канала DMA dmar_clear[1] - сбрасывает запрос 1-го канала DMA и т.д.
15	dmar_control_disable	Отключение режима контроля DMAR 0 - контроль включен. Запросы сохраняются в случае неготовности DMA. 1 - контроль выключен. Запросы подаются на DMA без дополнительной обработки
14	-	Резерв
13:11	arprot_control	Управление значением на порте arprot
10:7	arcache_control	Управление значением на порте arcache
6:4	awprot_control	Управление значением на порте awprot
3:0	awcache_control	Управление значением на порте awcache

6.3.22 Регистр sdr_dma1_ctl

Регистр sdr_dma1_ctl.

Таблица 6.24. Формат регистра sdr_dma1_ctl

Номер бита	Условное обозначение	Назначение
31:24	dmar_status	Статус запросов DMA Сброс осуществляется записью "1". После записи автоматически сбрасывается в "0" dmar_status[0] - статус запроса 0-го канала DMA dmar_status[1] - статус запроса 1-го канала DMA и т.д.
23:16	dmar_clear	Сброс запросов DMA Сброс осуществляется записью "1". После записи автоматически сбрасывается в "0" dmar_clear[0] - сбрасывает запрос 0-го канала DMA dmar_clear[1] - сбрасывает запрос 1-го канала DMA и т.д.
15	dmar_control_disable	Отключение режима контроля DMAR 0 - контроль включен. Запросы сохраняются в случае неготовности DMA. 1 - контроль выключен. Запросы подаются на DMA без дополнительной обработки
14	-	Резерв
13:11	arprot_control	Управление значением на порте arprot
10:7	arcache_control	Управление значением на порте arcache
6:4	awprot_control	Управление значением на порте awprot
3:0	awcache_control	Управление значением на порте awcache

6.3.23 Регистр sdr_dsp0_ppolicy

Регистр sdr_dsp0_ppolicy.

Таблица 6.25. Формат регистра sdr_dsp0_ppolicy

Номер бита	Условное обозначение	Назначение
31:5	-	Резерв
4:0	policy	Установка состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; 'b00001 PP_OFF - выключено; другое - резерв/выключено

6.3.24 Регистр sdr_dsp0_pstatus

Регистр sdr_dsp0_pstatus.

Таблица 6.26. Формат регистра sdr_dsp0_pstatus

Номер бита	Условное обозначение	Назначение
31:5	-	Резерв
4:0	pstatus	Поле отражает текущую активную установку состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; 'b00001 PP_OFF - выключено; другое - резерв/выключено

6.3.25 Регистр sdr_dsp1_ppolicy

Регистр sdr_dsp1_ppolicy.

Таблица 6.27. Формат регистра sdr_dsp1_ppolicy

Номер бита	Условное обозначение	Назначение
31:5	-	Резерв
4:0	policy	Установка состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; 'b00001 PP_OFF - выключено; другое - резерв/выключено

6.3.26 Регистр sdr_dsp1_pstatus

Регистр sdr_dsp1_pstatus.

Таблица 6.28. Формат регистра sdr_dsp1_pstatus

Номер бита	Условное обозначение	Назначение
31:5	-	Резерв

Номер бита	Условное обозначение	Назначение
4:0	pstatus	Поле отражает текущую активную установку состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; 'b00001 PP_OFF - выключено; другое - резерв/выключено

6.3.27 Регистр sdr_risc1_ppolicy

Регистр sdr_risc1_ppolicy.

Таблица 6.29. Формат регистра sdr_risc1_ppolicy

Номер бита	Условное обозначение	Назначение
31:5	-	Резерв
4:0	policy	Установка состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; 'b00001 PP_OFF - выключено; другое - резерв/выключено

6.3.28 Регистр sdr_risc1_pstatus

Регистр sdr_risc1_pstatus.

Таблица 6.30. Формат регистра sdr_risc1_pstatus

Номер бита	Условное обозначение	Назначение
31:5	-	Резерв
4:0	pstatus	Поле отражает текущую активную установку состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; 'b00001 PP_OFF - выключено; другое - резерв/выключено

6.3.29 Регистр sdr_acc_rst_ctl

Регистр sdr_acc_rst_ctl.

Таблица 6.31. Формат регистра sdr_acc_rst_ctl

Номер бита	Условное обозначение	Назначение
31:3	-	Резерв
2	acc2_sw_reset	Программный сброс. Активный уровень 0 ACC2
1	acc1_sw_reset	Программный сброс. Активный уровень 0 ACC1
0	acc0_sw_reset	Программный сброс. Активный уровень 0 ACC0

6.3.30 Регистр sdr_gnss_rst_ctl

Регистр sdr_gnss_rst_ctl.

Таблица 6.32. Формат регистра sdr_gnss_rst_ctl

Номер бита	Условное обозначение	Назначение
31:2	-	Резерв
1	cmos_sw_reset	Программный сброс. Активный уровень 0
0	sw_reset	Программный сброс. Активный уровень 0

6.3.31 Регистр sdr_a_dfe_rst_ctl

Регистр sdr_a_dfe_rst_ctl.

Таблица 6.33. Формат регистра sdr_a_dfe_rst_ctl

Номер бита	Условное обозначение	Назначение
31:1	-	Резерв
0	sw_reset	Программный сброс. Активный уровень 0

6.3.32 Регистр sdr_jesd0_rst_ctl

Регистр sdr_jesd0_rst_ctl.

Таблица 6.34. Формат регистра sdr_jesd0_rst_ctl

Номер бита	Условное обозначение	Назначение
31:2	-	Резерв
1	sw_reset_csr	Программный сброс. Активный уровень 0
0	sw_reset_core	Программный сброс. Активный уровень 0

6.3.33 Регистр sdr_jesd1_rst_ctl

Регистр sdr_jesd1_rst_ctl.

Таблица 6.35. Формат регистра sdr_jesd1_rst_ctl

Номер бита	Условное обозначение	Назначение
31:2	-	Резерв
1	sw_reset_csr	Программный сброс. Активный уровень 0
0	sw_reset_core	Программный сброс. Активный уровень 0

6.3.34 Регистр sdr_n_dfe_rst_ctl

Регистр sdr_n_dfe_rst_ctl.

Таблица 6.36. Формат регистра sdr_n_dfe_rst_ctl

Номер бита	Условное обозначение	Назначение
31:1	-	Резерв
0	sw_reset	Программный сброс. Активный уровень 0

6.3.35 Регистр sdr_ext_rst_ctl

Регистр sdr_ext_rst_ctl.

Таблица 6.37. Формат регистра sdr_ext_rst_ctl

Номер бита	Условное обозначение	Назначение
31:1	-	Резерв
0	sw_reset	Программный сброс. Активный уровень 0

6.3.36 Регистр sdr_bbd_rst_ctl

Регистр sdr_bbd_rst_ctl.

Таблица 6.38. Формат регистра sdr_bbd_rst_ctl

Номер бита	Условное обозначение	Назначение
31:1	-	Резерв
0	sw_reset	Программный сброс. Активный уровень 0

6.3.37 Регистр sdr_pci_rst_ctl

Регистр sdr_pci_rst_ctl.

Таблица 6.39. Формат регистра sdr_pci_rst_ctl

Номер бита	Условное обозначение	Назначение
31:1	-	Резерв
0	sw_reset	Программный сброс. Активный уровень 0

6.3.38 Регистр SDR_TRUST_ZONE

Регистр SDR_TRUST_ZONE.

Таблица 6.40. Формат регистра SDR_TRUST_ZONE

Номер бита	Условное обозначение	Назначение
31:1	-	Резерв
0	enable	Управление доступом в адресное пространство SDR со стороны системы 1 - доступ запрещен 0 - доступ разрешен

6.3.39 Регистр DFE_TRUST_ZONE

Регистр DFE_TRUST_ZONE.

Таблица 6.41. Формат регистра DFE_TRUST_ZONE

Номер бита	Условное обозначение	Назначение
31:1	-	Резерв

Номер бита	Условное обозначение	Назначение
0	enable	Управление проверкой данных в DFE 1 - проверка включена 0 - проверка выключена

6.3.40 Регистр LVDS_SEG1_CTL

Регистр LVDS_SEG1_CTL.

Таблица 6.42. Формат регистра LVDS_SEG1_CTL

Номер бита	Условное обозначение	Назначение
31:18	-	Резерв
17	PWRDNX	Управление режимом потребления КП сегмента SEG1 1 – режим пониженного потребления выключен 0 – режим пониженного потребления включен
16	PWRDN	Управление режимом потребления КП сегмента SEG1 0 – режим пониженного потребления выключен 1 – режим пониженного потребления включен
15:13	-	Резерв
12	RTT_PAD_EN	Сигнал включения RTT_PAD 0 – выключен 1 – включен
11:9	-	Резерв
8:5	RTT_COMP_BYP_R	Значение компенсации
4	RTT_COMP_BYP_EN	Перевод RTT_COMP в режим байпас 0 – байпас выключен 1 – байпас включен
3	RTT_COMP_UPDATE_EN	Сигнал разрешения перехода на новое значение компенсации
2	RTT_COMP_MVG_EN	В некоторых случаях значение компенсации может сильно изменяться относительно предыдущего. Чтобы не допустить скачка настроек и не вызвать значительного скачка импеданса следует установить «1» в данном поле. В данном режиме значение компенсации будет изменяться на единицу относительно предыдущего.
1	RTT_COMP_EN	Сигнал для включения компенсации 0 – компенсация выключена 1 – компенсация включена
0	RTT_COMP_POR	Сигнал сброса КП RTT_COMP сегмента SEG1 Активный уровень – «1»

6.3.41 Регистр LVDS_SEG1_STATUS

Регистр LVDS_SEG1_STATUS.

Таблица 6.43. Формат регистра LVDS_SEG1_STATUS

Номер бита	Условное обозначение	Назначение
31:25	-	Резерв
8:5	RTT_COMP_RBC	Значение компенсации
4	-	Резерв
3	RTT_COMP_UPDT	Признак завершения процедуры компенсации

Номер бита	Условное обозначение	Назначение
2:0	-	Резерв

6.3.42 Регистр LVDS_SEG1_RX_EN_CTL

Регистр включения КП LVDS_SEG1_RX_EN_CTL.

Таблица 6.44. Формат регистра LVDS_SEG1_RX_EN_CTL

Номер бита	Условное обозначение	Назначение
31:24	-	Резерв
23	SPARE_LVDS_RX_EN	0 – выключено / 1 – включено
22	JESD1_CLK_EN	0 – выключено / 1 – включено
21	JESD0_CLK_EN	0 – выключено / 1 – включено
20	DAC1_CLK_EN	0 – выключено / 1 – включено
19	ADC2_CLK_EN	0 – выключено / 1 – включено
18	ADC2_FRAME_EN	Общая КП для ADC2_FRAME и JESD1_SYNCB_I 0 – выключено / 1 – включено
17	ADC2_D7_EN	0 – выключено / 1 – включено
16	ADC2_D6_EN	0 – выключено / 1 – включено
15	ADC2_D5_EN	0 – выключено / 1 – включено
14	ADC2_D4_EN	0 – выключено / 1 – включено
13	ADC2_D3_EN	0 – выключено / 1 – включено
12	ADC2_D2_EN	0 – выключено / 1 – включено
11	ADC2_D1_EN	0 – выключено / 1 – включено
10	ADC2_D0_EN	0 – выключено / 1 – включено
9	ADC1_D7_EN	0 – выключено / 1 – включено
8	ADC1_D6_EN	0 – выключено / 1 – включено
7	ADC1_D5_EN	0 – выключено / 1 – включено
6	ADC1_D4_EN	0 – выключено / 1 – включено
5	ADC1_D3_EN	0 – выключено / 1 – включено
4	ADC1_D2_EN	0 – выключено / 1 – включено
3	ADC1_D1_EN	0 – выключено / 1 – включено
2	ADC1_D0_EN	0 – выключено / 1 – включено
1	ADC1_FRAME_EN	0 – выключено / 1 – включено
0	ADC1_CLK_EN	0 – выключено / 1 – включено

6.3.43 Регистр LVDS_SEG1_RX_HYST_EN_CTL

Регистр включения гистерезиса LVDS_SEG1_RX_HYST_EN_CTL.

Таблица 6.45. Формат регистра LVDS_SEG1_RX_HYST_EN_CTL

Номер бита	Условное обозначение	Назначение
31:24	-	Резерв
23	SPARE_LVDS_RX_HYST_EN	0 – выключено / 1 – включено
22	JESD1_CLK_HYST_EN	0 – выключено / 1 – включено
21	JESD0_CLK_HYST_EN	0 – выключено / 1 – включено

Номер бита	Условное обозначение	Назначение
20	DAC1_CLK_HYST_EN	0 – выключено / 1 – включено
19	ADC2_CLK_HYST_EN	0 – выключено / 1 – включено
18	ADC2_FRAME_HYST_EN	0 – выключено / 1 – включено
17	ADC2_D7_HYST_EN	0 – выключено / 1 – включено
16	ADC2_D6_HYST_EN	0 – выключено / 1 – включено
15	ADC2_D5_HYST_EN	0 – выключено / 1 – включено
14	ADC2_D4_HYST_EN	0 – выключено / 1 – включено
13	ADC2_D3_HYST_EN	0 – выключено / 1 – включено
12	ADC2_D2_HYST_EN	0 – выключено / 1 – включено
11	ADC2_D1_HYST_EN	0 – выключено / 1 – включено
10	ADC2_D0_HYST_EN	0 – выключено / 1 – включено
9	ADC1_D7_HYST_EN	0 – выключено / 1 – включено
8	ADC1_D6_HYST_EN	0 – выключено / 1 – включено
7	ADC1_D5_HYST_EN	0 – выключено / 1 – включено
6	ADC1_D4_HYST_EN	0 – выключено / 1 – включено
5	ADC1_D3_HYST_EN	0 – выключено / 1 – включено
4	ADC1_D2_HYST_EN	0 – выключено / 1 – включено
3	ADC1_D1_HYST_EN	0 – выключено / 1 – включено
2	ADC1_D0_HYST_EN	0 – выключено / 1 – включено
1	ADC1_FRAME_HYST_EN	0 – выключено / 1 – включено
0	ADC1_CLK_HYST_EN	0 – выключено / 1 – включено

6.3.44 Регистр LVDS_SEG2_CTL

Регистр LVDS_SEG2_CTL.

Таблица 6.46. Формат регистра LVDS_SEG2_CTL

Номер бита	Условное обозначение	Назначение
31:18	-	Резерв
17	REF_PWRDNX	Управление режимом потребления КП сегмента SEG2 1 – режим пониженного потребления выключен 0 – режим пониженного потребления включен

Номер бита	Условное обозначение	Назначение
16	REF_PWRDN	Управление режимом потребления КП сегмента SEG2 0 – режим пониженного потребления выключен 1 – режим пониженного потребления включен
15:0	-	Резерв

6.3.45 Регистр LVDS_SEG2_TX_OE_CTL

Регистр LVDS_SEG2_TX_OE_CTL.

Таблица 6.47. Формат регистра LVDS_SEG2_RT_OE_CTL

Номер бита	Условное обозначение	Назначение
31:18	-	Резерв
17	SPARE_LVDS_TX_OE	0 – tri_state 1 – enable
16	JESD_SYNCB_O_OE	0 – tri_state 1 – enable
15	DAC1_D15_OE	0 – tri_state 1 – enable
14	DAC1_D14_OE	0 – tri_state 1 – enable
13	DAC1_D13_OE	0 – tri_state 1 – enable
12	DAC1_D12_OE	0 – tri_state 1 – enable
11	DAC1_D11_OE	0 – tri_state 1 – enable
10	DAC1_D10_OE	0 – tri_state 1 – enable
9	DAC1_D9_OE	0 – tri_state 1 – enable
8	DAC1_D8_OE	0 – tri_state 1 – enable
7	DAC1_D7_OE	0 – tri_state 1 – enable
6	DAC1_D6_OE	0 – tri_state 1 – enable
5	DAC1_D5_OE	0 – tri_state 1 – enable
4	DAC1_D4_OE	0 – tri_state 1 – enable
3	DAC1_D3_OE	0 – tri_state 1 – enable
2	DAC1_D2_OE	0 – tri_state 1 – enable
1	DAC1_D1_OE	0 – tri_state 1 – enable
0	DAC1_D0_OE	0 – tri_state 1 – enable

6.3.46 Регистр LVDS_SEG3_CTL

Регистр LVDS_SEG3_CTL.

Таблица 6.48. Формат регистра LVDS_SEG3_CTL

Номер бита	Условное обозначение	Назначение
31:18	-	Резерв
17	PWRDNX	Управление режимом потребления КП сегмента SEG3 1 – режим пониженного потребления выключен 0 – режим пониженного потребления включен
16	PWRDN	Управление режимом потребления КП сегмента SEG3 0 – режим пониженного потребления выключен 1 – режим пониженного потребления включен
15:13	-	Резерв
12	RTT_PAD_EN	Сигнал включения RTT_PAD 0 – выключен 1 – включен
11:9	-	Резерв
8:5	RTT_COMP_BYP_R	Значение компенсации
4	RTT_COMP_BYP_EN	Перевод RTT_COMP в режим байпас 0 – байпас выключен 1 – байпас включен
3	RTT_COMP_UPDATE_EN	Сигнал разрешения перехода на новое значение компенсации
2	RTT_COMP_MVG_EN	В некоторых случаях значение компенсации может сильно изменяться относительно предыдущего. Чтобы не допустить скачка настроек и не вызвать значительного скачка импеданса следует установить «1» в данном поле. В данном режиме значение компенсации будет изменяться на единицу относительно предыдущего.
1	RTT_COMP_EN	Сигнал для включения компенсации 0 – компенсация выключена 1 – компенсация включена
0	RTT_COMP_POR	Сигнал сброса КП RTT_COMP сегмента SEG3 Активный уровень – «1»

6.3.47 Регистр LVDS_SEG3_STATUS

Регистр LVDS_SEG3_STATUS.

Таблица 6.49. Формат регистра LVDS_SEG3_STATUS

Номер бита	Условное обозначение	Назначение
31:9	-	Резерв
8:5	RTT_COMP_RBC	Значение компенсации
4	-	Резерв
3	RTT_COMP_UPDT	Признак завершения процедуры компенсации
2:0	-	Резерв

6.3.48 Регистр LVDS_SEG3_RX_EN_CTL

Регистр включения КП LVDS_SEG3_RX_EN_CTL.

Таблица 6.50. Формат регистра LVDS_SEG3_RX_EN_CTL

Номер бита	Условное Обозначение	Назначение
31:3	-	Резерв
2	DFE_CLK_EN	0 – выключено / 1 – включено
1	JESD1_SYSREF_EN	0 – выключено / 1 – включено
0	JESD0_SYSREF_EN	0 – выключено / 1 – включено

6.3.49 Регистр LVDS_SEG3_RX_HYST_EN_CTL

Регистр включения гистерезиса LVDS_SEG3_RX_HYST_EN_CTL.

Таблица 6.51. Формат регистра LVDS_SEG3_RX_HYST_EN_CTL

Номер бита	Условное обозначение	Назначение
31:3	-	Резерв
2	DFE_CLK_HYST_EN	0 – выключено / 1 – включено
1	JESD1_CLK_HYST_EN	0 – выключено / 1 – включено
0	JESD0_SYSREF_HYST_EN	0 – выключено / 1 – включено

6.3.50 Регистр CMOS_GPIO_REN_CTL

Регистр управления подтяжкой к земле/питанию CMOS_GPIO_REN_CTL.

Таблица 6.52. Формат регистра CMOS_GPIO_REN_CTL

Номер бита	Условное обозначение	Назначение
31:0	DFE_GPIO_31_0_REN	Каждый бит регистра отвечает за включение подтяжки к земле соответствующей ему КП [0] – КП DFE_GPIO[0] [1] – КП DFE_GPIO[1] [2] – КП DFE_GPIO[1] и т.д. 0 – подтяжка включена 1 – подтяжка выключена

6.3.51 Регистр CMOS_MIX_REN_CTL

Регистр управления подтяжкой к земле/питанию CMOS_MIX_REN_CTL.

Таблица 6.53. Формат регистра CMOS_MIX_REN_CTL

Номер бита	Условное обозначение	Назначение
31:23	-	Резерв
22:11	GNSS_CMOS_DIN_REN	0 – подтяжка к земле включена 1 – подтяжка к земле выключена
10	GNSS_CMOS_CLK_REN	0 – подтяжка к земле включена 1 – подтяжка к земле выключена

Номер бита	Условное обозначение	Назначение
9	SPARE_CMOS1_REN	0 – подтяжка к земле включена 1 – подтяжка к земле выключена
8	SPARE_CMOS0_REN	0 – подтяжка к земле включена 1 – подтяжка к земле выключена
7	DFE_PPS_REN	0 – подтяжка к земле включена 1 – подтяжка к земле выключена
6	DFE_TIMER_REN	0 – подтяжка к земле включена 1 – подтяжка к земле выключена
5	DFE_DAC3_CLK_REN	0 – подтяжка к земле включена 1 – подтяжка к земле выключена
4	DFE_ADC4_CLK_REN	0 – подтяжка к земле включена 1 – подтяжка к земле выключена
3	PCI1_APP_LTSSM_REN	0 – подтяжка к земле включена 1 – подтяжка к земле выключена
2	PCI1_PERSTN_REN	0 – подтяжка к питанию включена 1 – подтяжка к питанию выключена
1	PCI0_APP_LTSSM_REN	0 – подтяжка к земле включена 1 – подтяжка к земле выключена
0	PCI0_PERSTN_REN	0 – подтяжка к питанию включена 1 – подтяжка к питанию выключена

6.3.52 Регистр SOFT_NMI_SET

Регистр SOFT_NMI_SET.

Таблица 6.54. Формат регистра SOFT_NMI_SET

Номер бита	Условное обозначение	Назначение
31:1	-	Резерв
0	SET	Запись любого значения приводит к установке прерывания NMI RISC1

6.3.53 Регистр SOFT_NMI_CLEAR

Регистр SOFT_NMI_CLEAR.

Таблица 6.55. Формат регистра SOFT_NMI_CLEAR

Номер бита	Условное обозначение	Назначение
31:1	-	Резерв
0	CLEAR	Запись любого значения приводит к снятию прерывания NMI RISC1

6.3.54 Регистр SOFT_NMI_MASK

Регистр SOFT_NMI_MASK.

Таблица 6.56. Формат регистра SOFT_NMI_MASK

Номер бита	Условное обозначение	Назначение
31:1	-	Резерв

Номер бита	Условное обозначение	Назначение
0	MASK	Маска прерывания NMI RISC1 1 – прерывание разрешено 0 – прерывание запрещено

6.3.55 Регистр SOFT_NMI_STATUS

Регистр SOFT_NMI_STATUS.

Таблица 6.57. Формат регистра SOFT_NMI_STATUS

Номер бита	Условное обозначение	Назначение
31:1	-	Резерв
0	STATUS	Немаскированный статус прерывания NMI RISC1

6.3.56 Регистр PCIE_PERSTn_CTL

Регистр PCIE_PERSTn_CTL.

Таблица 6.58. Формат регистра PCIE_PERSTn_CTL

Номер бита	Условное обозначение	Назначение
31:10	-	Резерв
9	PCI1_PERSTn	Управление сигналом сброса PERSTn блока PCIe1. Активный уровень "0".
8	PCI1_PERSTn_MODE	Режим подачи сигнала сброса PERSTn блока PCIe1 «1» - внешний мастер «0» - программный контроль
7:2	-	Резерв
1	PCI0_PERSTn	Управление сигналом сброса PERSTn блока PCIe0. Активный уровень "0".
0	PCI0_PERSTn_MODE	Режим подачи сигнала сброса PERSTn блока PCIe0 «1» - внешний мастер «0» - программный контроль

6.4 Функционирование

6.4.1 Синхронизация и сброс

6.4.1.1 Синхронизация

Подсистема SDR укомплектована тремя блоками PLL и восьмью делителями частоты типа UCG. Структурные схема представлена на Рисунок 6.3, Рисунок 6.4 и Рисунок 6.5.

Управление блоками PLL осуществляется с помощью регистров «sdr_pll*_ctl». После выхода из аппаратного сброса все PLL выключены. В этом состоянии на выход PLL подается входная референсная частота 27.456 МГц. Включение PLL производится записью коэффициента умножение референсной частоты в поле «pll_sel» регистра «sdr_pll*_ctl».

PLL формирует референсную частоту для блока деления частоты UCG, который в свою очередь формирует рабочие частоты блоков подсистемы.

Блок UCG имеет возможность формировать опорные частоты для блоков подсистемы из одной из четырех заведенных на него референсных частот. Выбор используемой референсной частоты осуществляется с помощью регистров «sdr_ucg*_ctl». Также эти регистры позволяют выполнить программный сброс блоков UCG. Исключением является UCG0. В нем нет возможности менять референсную частоту и осуществлять программный сброс, т.к. от него тактируется конфигурационный домен. Подробное описание работы блока UCG представлено в Главе 43 «Универсальный блок формирования тактовой частоты (UCG)».

Ниже представлен список получаемых в подсистеме SDR частот

От UCG0 (референсная частота PLL0):

0. Clk_cfg – частотный домен конфигурационного интерконнекта, программно доступных регистров всех блоков подсистемы и интерфейса (порта) «sdr_sys*». Этот домен должен оставаться включенным всегда, т.к. его отключение приведет к зависанию подсистемы с последующим восстановлением только через аппаратный сброс.

1. Ext_clk – частотный домен интерконнекта транслирующего запросы от подсистемы SDR во внешнее адресное пространство и интерфейсов (портов) «sdr_dev*» и «sdr_ddr*».

2. Int_clk – частотный домен внутреннего высокоскоростного интерконнекта осуществляющего передачу большого объема обрабатываемых в подсистеме данных и блоков «MPSM», «DMA*».

3. Pci_clk – частотный домен интерконнекта транслирующего запросы от блоков PCI во внешнее адресное пространство, интерфейсов (портов) «pci_dev*» и «pci_ddr*», блоков PCI0 и PCI1.

4. Vcu_clk – частотный домен блока VCU

5. Acc0_clk – частотный домен ускорителей «VITDEC» и «CONVENC»

6. Acc1_clk – частотный домен ускорителей «TDC» и «TEC»

7. Acc2_clk – частотный домен ускорителя «FFT»

8. Aux_pci_clk – вспомогательная частота для блоков «PCI*»

9. Gnss_clk – частота блока «Navicore_5l»

10. Dfe_alt_clk – частота блока «DFE_ALT»

11. Vcu_tclk – вспомогательная частота для блока «Vcu»

12. Lvds_clk – частота для тактирования контактных площадок типа LVDS

От UCG_PCI0_REF (референсные частоты PLL0, Jesd0_dev_clk, Jesd1_dev_clk):

0. Ref_alt_clk – референсная частота для блока «PCI0»

От UCG_JESD0_TX (референсные частоты Jesd0_dev_clk, Jesd1_dev_clk, Jesd0_tx_serdes_clk):

1. Jesd0_tx_sample_clk – тактирование блока Jesd0
2. Jesd0_tx_character_clk – тактирование блока Jesd0

От UCG_JESD0_RX (референсные частоты Jesd0_dev_clk, Jesd1_dev_clk, Jesd0_rx_serdes_clk):

1. Jesd0_rx_sample_clk – тактирование блока Jesd0
2. Jesd0_rx_character_clk – тактирование блока Jesd0

От UCG_N_DFE (референсные частоты PLL1, dfe_ext_clk, dfe_adc1_clk, dfe_dac1_clk):

0. Dfe_nels_clk – частота блока «DFE_NELS»

От UCG_PCI1_REF (референсные частоты PLL0, Jesd0_dev_clk, Jesd1_dev_clk):

0. Ref_alt_clk – референсная частота для блока «PCI0»

От UCG_JESD1_TX (референсные частоты Jesd0_dev_clk, Jesd1_dev_clk, Jesd1_tx_serdes_clk):

1. Jesd1_tx_sample_clk – тактирование блока Jesd1
2. Jesd1_tx_character_clk – тактирование блока Jesd1

От UCG_JESD1_RX (референсные частоты Jesd0_dev_clk, Jesd1_dev_clk, Jesd1_rx_serdes_clk):

1. Jesd1_rx_sample_clk – тактирование блока Jesd1
2. Jesd1_rx_character_clk – тактирование блока Jesd1

Синхросигнал на ядра DSP формируется на PLL2 без участия UCG.

6.4.1.2 Сброс

Сброс всех блоков подсистемы осуществляется аппаратно (сигналом HW_RSTn) и программно. Реализован отдельный программный сброс практически для каждого блока. После снятия аппаратного сброса большинство сигналов программного сброса находятся в активном состоянии и требуют переключения для перевода блока в рабочее состояние.

6.4.1.3 Управление сигналами синхронизации и сброса.

В таблице перечислены основные блоки подсистемы и соответствующие им регистры управления синхросигналами и сбросом.

Таблица 6.59.

Блок	Регистра управления частотой	Регистра управления сбросом	Комментарий
PLL0	sdr_pll0_ctl, sdr_pll0_diag	-	Блок умножения частоты PLL0 работает от референсной частоты 27.456 МГц. После снятия аппаратного сброса работает в режиме «байпас». Не имеет программного сброса.
UCG0	-	-	Блок деления и управления частотами UCG0 всегда тактируется частотой, сгенерированной на PLL0. После снятия аппаратного сброса пропускает входную частоту без деления. Не имеет программного сброса.
UCG_PCI0_REF	sdr_ucg_pci0_ref_ctl[1:0]	sdr_ucg_pci0_ref_ctl[8]	Блок деления и управления частотами UCG_PCI0_REF. Есть возможность выбора референсной частоты. После снятия аппаратного сброса находится под активным программным сбросом.
UCG_PCI1_REF	sdr_ucg_pci1_ref_ctl[1:0]	sdr_ucg_pci1_ref_ctl[8]	Блок деления и управления частотами UCG_PCI1_REF. Есть возможность выбора референсной частоты. После снятия аппаратного сброса находится под активным программным сбросом.
UCG_JESD0_TX	sdr_ucg_jesd0_tx_ref_ctl[1:0]	sdr_ucg_jesd0_tx_ref_ctl[8]	Блок деления и управления частотами UCG_JESD0_TX. Есть возможность выбора референсной частоты. После снятия аппаратного сброса находится под активным программным сбросом.
UCG_JESD0_RX	sdr_ucg_jesd0_rx_ref_ctl[1:0]	sdr_ucg_jesd0_rx_ref_ctl[8]	Блок деления и управления частотами UCG_JESD0_RX. Есть возможность выбора референсной частоты. После снятия аппаратного сброса находится под активным программным сбросом.
UCG_JESD1_TX	sdr_ucg_jesd1_tx_ref_ctl[1:0]	sdr_ucg_jesd1_tx_ref_ctl[8]	Блок деления и управления частотами UCG_JESD1_TX. Есть возможность выбора референсной частоты. После снятия аппаратного сброса находится под активным программным сбросом.
UCG_JESD1_RX	sdr_ucg_jesd1_rx_ref_ctl[1:0]	sdr_ucg_jesd1_rx_ref_ctl[8]	Блок деления и управления частотами UCG_JESD1_RX. Есть возможность выбора референсной частоты. После снятия аппаратного сброса находится под активным программным сбросом.

Системные регистры, порт sys*	Регистры управления 0-м каналом UCG0	-	Системная частота доступа к конфигурационным регистрам всех блоков формируется на 0-м канале блока UCG0. После снятия аппаратного сброса частота включена. **Выключать нельзя!** Не имеет программного сброса.
Канал доступа к внешним ресурсам (DDR, HSP и др)	Регистры управления 1-м каналом UCG0	sdr_ext_rst_ctl[0]	Частота доступа к внешним ресурсам формируется на 1-м канале блока UCG0. После снятия аппаратного сброса частота выключена. Блок находится под активным программным сбросом.
Канал доступа к внутренним ресурсам (MPSM, CRAM и др)	Регистры управления 2-м каналом UCG0	sdr_bbd_rst_ctl[0]	Частота доступа к внутренним ресурсам формируется на 2-м канале блока UCG0. После снятия аппаратного сброса частота выключена. Блок находится под активным программным сбросом.
PCIe0, PCIe1	Регистры управления 3-м каналом UCG0, sdr_pci*_ctl	sdr_pci_rst_ctl[0]	Частота работы контроллеров PCIe и доступа к внешним/внутренним ресурсам через интерконнект PCI формируется на 3-м канале блока UCG0. После снятия аппаратного сброса частота выключена. Блок находится под активным программным сбросом.
	Регистры управления 8-м каналом UCG0	-	Вспомогательная частота (aux_clk) работы контроллеров PCIe формируется на 8-м канале блока UCG0. После снятия аппаратного сброса частота включена. Не имеет программного сброса.
	Регистры управления 0-м каналом UCG_PCI*_REF, sdr_pci*_ctl	-	Референсная частота физического уровня PCIe формируется на 0-м канале блока UCG0_PCI*_REF. После снятия аппаратного сброса частота выключена. Не имеет программного сброса.
Jesd0, Jesd1	Регистры управления 1-м каналом UCG_JESD*_TX	sdr_jesd*_rst_ctl[1:0]	Частота tx_sample_clk контроллера Jesd* формируется на 1-м канале блока UCG_JESD*_TX. После снятия аппаратного сброса частота выключена. Блок находится под активным программным сбросом.
	Регистры управления 2-м каналом UCG_JESD*_TX		Частота tx_character_clk контроллера Jesd* формируется на 2-м канале блока UCG_JESD*_TX. После снятия аппаратного сброса частота выключена. Блок находится под активным программным сбросом.
	Регистры управления 1-м каналом UCG_JESD*_RX		Частота rx_sample_clk контроллера Jesd* формируется на 1-м канале блока UCG_JESD*_RX. После снятия аппаратного сброса частота выключена. Блок находится под активным программным сбросом.
	Регистры управления 2-м каналом UCG_JESD*_RX		Частота rx_character_clk контроллера Jesd* формируется на 2-м канале блока UCG_JESD*_RX. После снятия аппаратного сброса частота выключена. Блок находится под активным программным сбросом.
Risc1	Регистры управления 4-м каналом UCG0	sdr_risc1_ppolicy sdr_risc1_status	Частота работы Risc1 формируется на 4-м канале блока UCG0. После снятия аппаратного сброса частота выключена. Блок находится под активным программным сбросом.
	Регистры управления 11-м каналом UCG0		Частота работы Jtag Risc1 формируется на 11-м канале блока UCG0. После снятия аппаратного сброса частота выключена. Блок находится под активным программным сбросом.

VITDEC, CONVENC	Регистры управления 5-м каналом UCG0	sdr_acc_rst_ctl[0]	Частота работы блоков VITDEC и CONVENC формируется на 5-м канале блока UCG0. После снятия аппаратного сброса частота выключена. Блоки находятся под активным программным сбросом.
TDC, TEC	Регистры управления 6-м каналом UCG0	sdr_acc_rst_ctl[1]	Частота работы блоков TDC и TEC формируется на 6-м канале блока UCG0. После снятия аппаратного сброса частота выключена. Блоки находятся под активным программным сбросом.
FFT	Регистры управления 7-м каналом UCG0	sdr_acc_rst_ctl[2]	Частота работы блока FFT формируется на 7-м канале блока UCG0. После снятия аппаратного сброса частота выключена. Блок находится под активным программным сбросом.
Navicore_5l	Регистры управления 9-м каналом UCG0	sdr_gnss_rst_ctl[1:0]	Частота работы блока Navicore_5l формируется на 9-м канале блока UCG0. После снятия аппаратного сброса частота выключена. Программный сброс находится в неактивном.
DFE_ALT	Регистры управления 10-м каналом UCG0	sdr_a_dfe_rst_ctl[0]	Частота работы блока DFE_ALT формируется на 10-м канале блока UCG0. После снятия аппаратного сброса частота выключена. Блок находится под активным программным сбросом.
LVDS IO	Регистры управления 12-м каналом UCG0	LVDS_SEG1_CTL LVDS_SEG3_CTL	Частота настройки LVDS IO формируется на 12-м канале блока UCG0. После снятия аппаратного сброса частота выключена. Блок находится под активным программным сбросом.
PLL1	sdr_pll1_ctl, sdr_pll1_diag	-	Блок умножения частоты PLL1 работает от референсной частоты 27.456 МГц. После снятия аппаратного сброса работает в режиме «байпас». Не имеет программного сброса.
UCG_N_DFE	sdr_ucg_n_dfe_ctl[1:0]	sdr_ucg_n_dfe_ctl[8]	Блок деления и управления частотами UCG_N_DFE. Есть возможность выбора референсной частоты. После снятия аппаратного сброса находится под активным программным сбросом.
DFE_NELS	Регистры управления 0-м каналом UCG_N_DFE	sdr_n_dfe_rst_ctl[0]	Частота работы блока DFE_NELS формируется на 0-м канале блока UCG_N_DFE. После снятия аппаратного сброса частота выключена. Блок находится под активным программным сбросом.
PLL2	sdr_pll2_ctl, sdr_pll2_diag	-	Блок умножения частоты PLL2 работает от референсной частоты 27.456 МГц. После снятия аппаратного сброса работает в режиме «байпас». Не имеет программного сброса.
DSP0, DSP1	sdr_dsp_ctl[9:8] sdr_pll2_ctl, sdr_pll2_diag	sdr_dsp*_ppolicy sdr_dsp*_pstaus	Частота работы блока DSP* формируется на PLL2. После снятия аппаратного сброса частота выключена. Блок находится под активным программным сбросом.

6.4.2 Запуск RISC1

Для пуска процессора RISC1 необходимо выполнить следующие действия:

1. Включить частоты:
    a. CLK_AXI_INT (UCG0 2-й канал)
    b. CLK_VCU (UCG0 4-й канал)

c. VCU_TCLK (UCG0 8-й канал)

2. Перевести в неактивное состояние сигналы сброса:

   a. RSTn_AXI_INT (регистр sdr_bbd_rst_ctl)

   b. RSTn_VCU (регистр sdr_risc1_ppolicy)

3. Сгенерировать NMI (регистр SOFT_NMI_SET)

6.4.3 Зоны доступа и проверка данных

Область конфигурационных регистров (0x0190_0000 - 0x027F_FFFF) доступна всем.

Область накристальной памяти (0x0300_0000 - 0x03FF_FFFF) доступна всем устройствам SoC, если бит «enable» регистра «sdr_trust_zone» равен 0. Если «enable» равен 1, то доступ запрещается блокам PCIe и всем компонентам, размещенным за пределами подсистемы.

Для обеспечения защиты данных поступающих в блок DFE осуществляется их проверка. Для включения/отключения проверки реализован регистр «dfe_trust_zone».

6.4.4 Распределение запросов на прерывание подсистемы SDR

Таблица 6.60. Распределение запросов на прерывание

Источник запроса	Цель
QLIC1[0]	RISC1 IP[2]
QLIC1[1]	RISC1 IP[3]
QLIC1[2]	RISC1 IP[4]
QLIC1[3]	RISC1 IP[5]
soft_nmi (с SW регистра)	RISC1 [NMI]
QLIC1[5]	Не используется
QLIC1[6]	Не используется
QLIC1[7]	Не используется
QLIC1[8]	Не используется
QLIC1[9]	DSP0[IRQIN]
QLIC1[10]	DSP0[IRQIN_DBG]
QLIC1[11]	DSP1[IRQIN]
QLIC1[12]	DSP1[IRQIN_DBG]
QLIC1[13]	Не используется
QLIC1[14]	Не используется
QLIC1[15]	Не используется
QLIC1[16]	GIC500 PPI31 CPU0, 31 и SPI43
	GIC500 PPI31 CPU1, 31
	GIC500 PPI31 CPU2, 31
	GIC500 PPI31 CPU3, 31
QLIC1[17]	QLIC0[43]

6.4.5 Распределение событий

Обмен событиями реализован для двух групп блоков:

1. CPU, RISC0, RISC1, DSP0, DSP1
2. ACC, DMA0, DMA1

Распределение для первой группы:

Таблица 6.61.

Источник события	Приемник события 1	Приемник события 2	Приемник события 3
DSP1[0]	DSP0_EVENT_IN[0]	DSP1_EVENT_IN[0]	dsp1_event*
DSP1[1]	DSP0_EVENT_IN[1]	DSP1_EVENT_IN[1]	
DSP1[2]	DSP0_EVENT_IN[2]	DSP1_EVENT_IN[2]	
DSP1[3]	DSP0_EVENT_IN[3]	DSP1_EVENT_IN[3]	
DSP0[0]	DSP0_EVENT_IN[4]	DSP1_EVENT_IN[4]	dsp0_event*
DSP0[1]	DSP0_EVENT_IN[5]	DSP1_EVENT_IN[5]	
DSP0[2]	DSP0_EVENT_IN[6]	DSP1_EVENT_IN[6]	
DSP0[3]	DSP0_EVENT_IN[7]	DSP1_EVENT_IN[7]	
RISC1	DSP0_EVENT_IN[8]	DSP1_EVENT_IN[8]	risc1_event*
RISC0	DSP0_EVENT_IN[9]	DSP1_EVENT_IN[9]	
CPU	DSP0_EVENT_IN[10]	DSP1_EVENT_IN[10]	

* - дальнейшее распределение событий dsp0_event, dsp1_event, risc1_event описано в Главе 3 «Системная организация микросхемы».

Распределение для второй группы:

Таблица 6.62.

Источник события	Приемник события
vitdec_tx	Dma0_ndmar[7]
vitdec_rx	Dma0_ndmar[6]
cent_tx	Dma0_ndmar[5]
cent_rx	Dma0_ndmar[4]
tdc_rx0	Dma1_ndmar[7]
tdc_rx1	Dma1_ndmar[6]
tec_tx	Dma1_ndmar[5]
tec_rx	Dma1_ndmar[4]

6.4.5.1 Управление событиями

Для увеличения гибкости взаимодействия ускорителей с DMA предусмотрен механизм управления событиями. Данный механизм позволяет запоминать события пришедшие от ускорителей в момент, когда DMA не готов его принять и хранит его до тех пор, пока DMA его не примет или не будет сброшено программно.

Отключение этого механизма, анализ состояния сохраненных событий, а также их сброс осуществляется через регистры:

- sdr_dma0_ctl (0x0058)
- sdr_dma1_ctl (0x005C)

6.4.6 Особенности настройки частот Jesd204b

Для корректного функционирования блока необходимы следующие частоты:

- tx_clk_sample – частота приема потока данных от DFE (источник sdr_ucg_jesd_tx)
- tx_clk_character – частота обработки данных (источник sdr_ucg_jesd_tx)
- tx_clk_serdes – частота передачи данных на PHY (источник pcie_phy)
- rx_clk_sample – частота передачи потока данных на DFE (источник sdr_ucg_jesd_rx)
- rx_clk_character – частота обработки данных (источник sdr_ucg_jesd_rx)
- rx_clk_serdes[3:0] – частота приема данных с PHY (источник pcie_phy)

6.4.7 Управление контактными площадками

6.4.7.1 Мультиплексирование GPIO

В подсистеме SDR используется мультиплексирование контактных площадок общего назначения. Выбор режима (Функции) работы осуществляется при помощи регистра «Регистр sdr_n_dfe_ctl». Контактные площадки DFE_GPIO* можно использовать для следующих целей:

- В режиме «Функция 0» все 32 КП работают в режиме GPIO под управлением блока DFE_NELS
    - DFE_GPIO[31:0] – в режиме GPIO
- В режиме «Функция 1» также все КП работают под управлением блока DFE_NELS, но большинство из них используются для подключения внешних устройств по SSI интерфейсу. Реализовано 4 независимых SSI с возможность подключения до 4-х подканалов (до 15, если к портам *cs* подключить внешний дешифратор 1:2:4:8 в 16):
    - DFE_GPIO[6:0] – SSI1
    - DFE_GPIO[13:7] – SSI2
    - DFE_GPIO[15:14] – в режиме GPIO
    - DFE_GPIO[22:16] – SSI3
    - DFE_GPIO[29:23] – SSI4
    - DFE_GPIO[31:30] – в режиме GPIO

- В режиме «Функция 2» часть КП отдается для подключения 16-разрядного АЦП (с управляющим сигналом "frame"), при этом сохраняются два интерфейса SSI:
    - DFE_GPIO[7:0] – АЦП data Lo (заводится на порты dfe_nels_adc4_d[7:0] блока DFE_NELS и порт dfe_alt_adc_d[7:0] блока DFE_ALT)
    - DFE_GPIO[15:8] – АЦП data Hi (заводится на порты dfe_nels_adc5_d[7:0] блока DFE_NELS и порт dfe_alt_adc_d[15:8] блока DFE_ALT)
    - DFE_GPIO[30] – АЦП frame (заводится на порт dfe_nels_adc4_frame блока DFE_NELS)
    - DFE_GPIO[22:16] – SSI3
    - DFE_GPIO[29:23] – SSI4
    - DFE_GPIO[31] – в режиме GPIO
- В режиме «Функция 3» вместо интерфейсов SSI на КП выводятся отладочные сигналы от блоков DFE_ALT и GNSS:
    - DFE_GPIO[7:0] – АЦП data Lo (заводится на порты dfe_nels_adc4_d[7:0] блока DFE_NELS и порт dfe_alt_adc_d[7:0] блока DFE_ALT)
    - DFE_GPIO[15:8] – АЦП data Hi (заводится на порты dfe_nels_adc5_d[7:0] блока DFE_NELS и порт dfe_alt_adc_d[15:8] блока DFE_ALT)
    - DFE_GPIO[30] – АЦП frame (заводится на порт dfe_nels_adc4_frame блока DFE_NELS)
    - DFE_GPIO[23:16] – в режиме GNSS_GPIO (под управлением блока GNSS)
    - DFE_GPIO[29:24] – DFE_ALT_TST
    - DFE_GPIO[31] – DFE_ALT_TX_MARK

6.4.7.2 LVDS

6.4.7.2.1 Разделение на сегменты

В подсистеме реализованы 3 независимых сегмента содержащие КП типа LVDS:

1. SEG1. Содержит следующие КП:
    a. Функциональные:
        - DFE_ADC1*_PAD*

- DFE_ADC2*_PAD*
- JESD0_CLK_PAD*
- JESD1_CLK_PAD*
- SPARE_LVDS_RX_PAD*
- DFE_DAC1_CLK_PAD*

b. Служебные:

- SEG1_REF*_PAD
- SEG1_RTTPAD_PAD*

2. SEG2. Содержит следующие КП:

a. Функциональные:

- DFE_DAC1_D*_PAD*
- JESD_SYNCB_O_PAD*
- SPARE_LVDS_TX_PAD*

b. Служебные:

- SEG2_REF*_PAD

3. SEG3. Содержит следующие КП:

a. Функциональные:

- DFE_CLK_PAD*
- JESD0_SYSREF_PAD*
- JESD1_SYSREF_PAD*

b. Служебные:

- SEG3_REF*_PAD
- SEG3_RTTPAD_PAD*

6.4.7.2.2 Перевод в рабочее состояние

Для использования любой КП необходимо выполнить следующие действия:

1. Вывести соответствующий КП сегмент из состояния пониженного потребления (по сбросу все сегменты выведены из режима пониженного потребления) используя поля PWRDN/ PWRDNX регистров LVDS_SEG*_CTL.

2. Включить блок компенсации для обеспечения точного выходного сопротивления драйвера (только для сегментов SEG1 и SEG3).

6.4.7.2.3 Блок компенсации

Результатом работы блока компенсации является установка нужного значения на шине CTLH соединяющей все КП одного сегмента.

6.4.7.2.3.1 RTT компенсация

Блок RTT компенсации используется для обеспечения точного выходного сопротивления драйвера с использованием внешних эталонных резисторов.

Для его включения требуется:

1. Включить частоту CLK_LVDS (UCG0[12]) – не более 200МГц

2. Включить RTT_PAD (LVDS_SEG*_CTL[RTT_PAD_EN]) и блок компенсации (LVDS_SEG*_CTL[RTT_COMP_EN])

3. Сформировать положительный фронт сброса (LVDS_SEG*_CTL[RTT_COMP_POR) – записать последовательно 0/1/0

4. Дождаться появления «1» в регистре статуса LVDS_SEG*_STATUS[RTT_COMP_UPDT]. При этом в поле RTT_COMP_RBC регистра LVDS_SEG*_STATUS будет отображаться значение компенсации.

5. Сформировать импульс на сигнале разрешения обновления значения компенсации (LVDS_SEG*_CTL[RTT_COMP_UPDATE_EN]) – записать последовательно 1/0

6.4.7.2.3.2 Байпас

Если не удается выполнить процедуру RTT компенсации, то можно установить значение компенсации программно. Для этого:

1. Записать нужное значение компенсации в поле LVDS_SEG*_CTL[RTT_COMP_BYP_R]

2. Записать «1» в поле LVDS_SEG*_CTL[RTT_COMP_BYP_EN]. Включится режим байпас, при этом не важно, что выставлено LVDS_SEG*_CTL[RTT_COMP_EN]. Для компенсации будет использовано значение с поля RTT_COMP_BYP_R

6.4.7.3 CMOS

Контактные площадки CMOS типа не требуют дополнительного управления. К таким КП относятся:

- DFE_GPIO*
- DFE_DAC3*
- DFE_TIMER
- DFE_PPS
- DFE_ADC4_CLK
- GNSS*
- PCI*_PERSTN
- PCI*_WAKE
- PCI*_APP_LTSMM_EN
- PCIE*_CLKREQ

Для данных КП доступна лишь функция включения подтяжки к земле/питанию, которая осуществляется через регистры CMOS_GPIO_REN_CTL и CMOS_MIX_REN_CTL.

6.4.7.4 Выбор режима работы Jesd204b/PCIe

Внешние устройства использующие интерфейсы Jesd204b и PCIe подключаются через общие контактные площадки:

- JESD*_CLK_PAD
- JESD*_CLK_PADX
- JESD*_SYSREF_PAD
- JESD*_SYSREF_PADX
- ADC2_FRAME_PAD
- ADC2_FRAME_PADX
- JESD_SYNCB_O_PAD
- JESD_SYNCB_O_PADX

- SPARE_RX_PAD
- SPARE_RX_PADX
- SPARE_TX_PAD
- SPARE_TX_PADX
- PCI*_PERSTN_PAD
- PCI*_WAKE_PAD
- PCI*_APP_LTSMM_EN_PAD
- PCI*_CLKREQ_PAD
- PCI*_PHY_RESREF_PAD
- PCI*_REF_CLK_M_PAD
- PCI*_REF_CLK_P_PADX
- PCI*_RXN_PAD[3:0]
- PCI*_RXP_PADX[3:0]
- PCI*_TXN_PAD[3:0]
- PCI*_TXP_PADX[3:0]
- PCI*_REF_CLK_M_PAD
- PCI*_REF_CLK_P_PADX

Выбор режима работы осуществляется через программно доступный регистр (см. описание PCIe). В режиме работы Jesd204b не используются следующие КП:

- PCI*_PERSTN_PAD – может быть не подключен (NC), но должна быть включена внутренняя подтяжка к VDD (через SW регистр)
- PCI*_WAKE_PAD – может быть не подключен (NC)
- PCI*_APP_LTSMM_EN_PAD – может быть не подключен (NC), но должна быть включена внутренняя подтяжка к VSS (через SW регистр)
- PCI*_CLKREQ_PAD –может быть не подключен (NC)

В режиме работы PCIe не используются следующие КП:

- JESD*_CLK_PAD – может быть не подключен (NC), но должна быть выключена через SW регистр
- JESD*_CLK_PADX – может быть не подключен (NC), но должна быть выключена через SW регистр
- JESD*_SYSREF_PAD – может быть не подключен (NC), но должна быть выключена через SW регистр
- JESD*_SYSREF_PADX – может быть не подключен (NC), но должна быть выключена через SW регистр
- SPARE_RX_PAD – может быть не подключен (NC), но должна быть выключена через SW регистр
- SPARE_RX_PADX – может быть не подключен (NC), но должна быть выключена через SW регистр
- SPARE_TX_PAD – может быть не подключен (NC), но должна быть выключена через SW регистр
- SPARE_TX_PADX – может быть не подключен (NC), но должна быть выключена через SW регистр
- JESD_SYNCB_O_PAD – может быть не подключен (NC), но должна быть выключена через SW регистр
- JESD_SYNCB_O_PADX – может быть не подключен (NC), но должна быть выключена через SW регистр

6.4.8 Система отладки

Система отладки включает в себя следующие компоненты

- BPAM (BPAM_RISC1, BPAM_DSP0, BPAM_DSP1) – модуль отладки процессора RISC1, DSP0 и DSP1
- BUS_MON – модуль контроля системной шины
- STATUS_MON – модуль контроля состояния блока DFE

6.4.8.1 BPAM_RISC1

BPAM_RISC1 позволяет в режиме отладки осуществить доступ к регистрам RISC1 (в адресное пространство 0x0192_0000 - 0x0192_FFFF)

Таблица 6.63. Подключение сигналов кросс отладки к компоненту BPAM_RISC1

Источник события	Приемник события
ust_event_in[1:0]	{risc1_debug_o, risc1_de_out_o }
ust_event_out[1:0]	{risc1_break_dsp_i, risc1_de_in_i }
ust_event_out_ack[1:0]	{risc1_en_break_dsp_o, risc1_de_out_o }
ust_gpio	не используется

6.4.8.2 BPAM_DSP0

BPAM_DSP0 позволяет в режиме отладки осуществить доступ к регистрам DSP0 (в адресное пространство 0x0198_0000 - 0x019B_FFFF)

Таблица 6.64. Подключение сигналов кросс отладки к компоненту BPAM_DSP0

Источник события	Приемник события
ust_event_in[3:0]	dsp0_events_out_o[3:0]
ust_event_out	не используется
ust_event_out_ack	не используется
ust_gpio	не используется

6.4.8.3 BPAM_DSP1

BPAM_DSP1 позволяет в режиме отладки осуществить доступ к регистрам DSP1 (в адресное пространство 0x01C8_0000 - 0x01CB_FFFF)

Таблица 6.65. Подключение сигналов кросс отладки к компоненту BPAM_DSP0

Источник события	Приемник события
ust_event_in[3:0]	dsp1_events_out_o[3:0]
ust_event_out	не используется
ust_event_out_ack	не используется
ust_gpio	не используется

6.4.8.4 BUS_MON

BUS_MON является пассивным компонентом. Занимается исключительно отслеживанием трафика на шине PCI_DEV_S.

6.4.8.5 STATUS_MON

STATUS_MON позволяет собирать статистику работы блока «DFE_NELS» на основе информации с отладочных выводов «ndfe_ultrasoc_debug_valid_rs_w[3:0]», «ndfe_ultrasoc_debug_rs_w [31:0]»

Таблица 6.66. Подключение сигналов кросс отладки к компоненту STATUS_MON

Источник события	Приемник события
ust_sts_match_valid_i	(smon_ust_gpio_output_w[15] & ndfe_ultrasoc_debug_valid_rs_w[3]) | (smon_ust_gpio_output_w[14] & ndfe_ultrasoc_debug_valid_rs_w[2]) | (smon_ust_gpio_output_w[13] & ndfe_ultrasoc_debug_valid_rs_w[1]) | (smon_ust_gpio_output_w[12] & ndfe_ultrasoc_debug_valid_rs_w[0])
ust_sts_match_i[31:0]	ndfe_ultrasoc_debug_rs_w[31:0]

ust_sts_filter_valid_i	(smon_ust_gpio_output_w[11] & ndfe_ultrasoc_debug_valid_rs_w[3]) | (smon_ust_gpio_output_w[10] & ndfe_ultrasoc_debug_valid_rs_w[2]) | (smon_ust_gpio_output_w[9] & ndfe_ultrasoc_debug_valid_rs_w[1]) | (smon_ust_gpio_output_w[8] & ndfe_ultrasoc_debug_valid_rs_w[0])
ust_sts_filter_i[31:0]	ndfe_ultrasoc_debug_rs_w[31:0]
ust_sts_data_valid_i	(smon_ust_gpio_output_w[7] & ndfe_ultrasoc_debug_valid_rs_w[3]) | (smon_ust_gpio_output_w[6] & ndfe_ultrasoc_debug_valid_rs_w[2]) | (smon_ust_gpio_output_w[5] & ndfe_ultrasoc_debug_valid_rs_w[1]) | (smon_ust_gpio_output_w[4] & ndfe_ultrasoc_debug_valid_rs_w[0])
ust_sts_data_i[31:0]	ndfe_ultrasoc_debug_rs_w[31:0]
ust_event_input_i[3:0]	{(smon_ust_gpio_output_w[3] & ndfe_ultrasoc_debug_valid_rs_w[3]), (smon_ust_gpio_output_w[2] & ndfe_ultrasoc_debug_valid_rs_w[2]), (smon_ust_gpio_output_w[1] & ndfe_ultrasoc_debug_valid_rs_w[1]), (smon_ust_gpio_output_w[0] & ndfe_ultrasoc_debug_valid_rs_w[0])}
ust_sts_accum_i[31:0]	ndfe_ultrasoc_debug_rs_w[31:0]
ust_gpio_output_o[15:0]	smon_ust_gpio_output_w[15:0]

7. ПОДСИСТЕМА MEDIA

7.1 Структурная схема

Структурная схема мультимедийной подсистемы представлена на Рисунок 7.1.

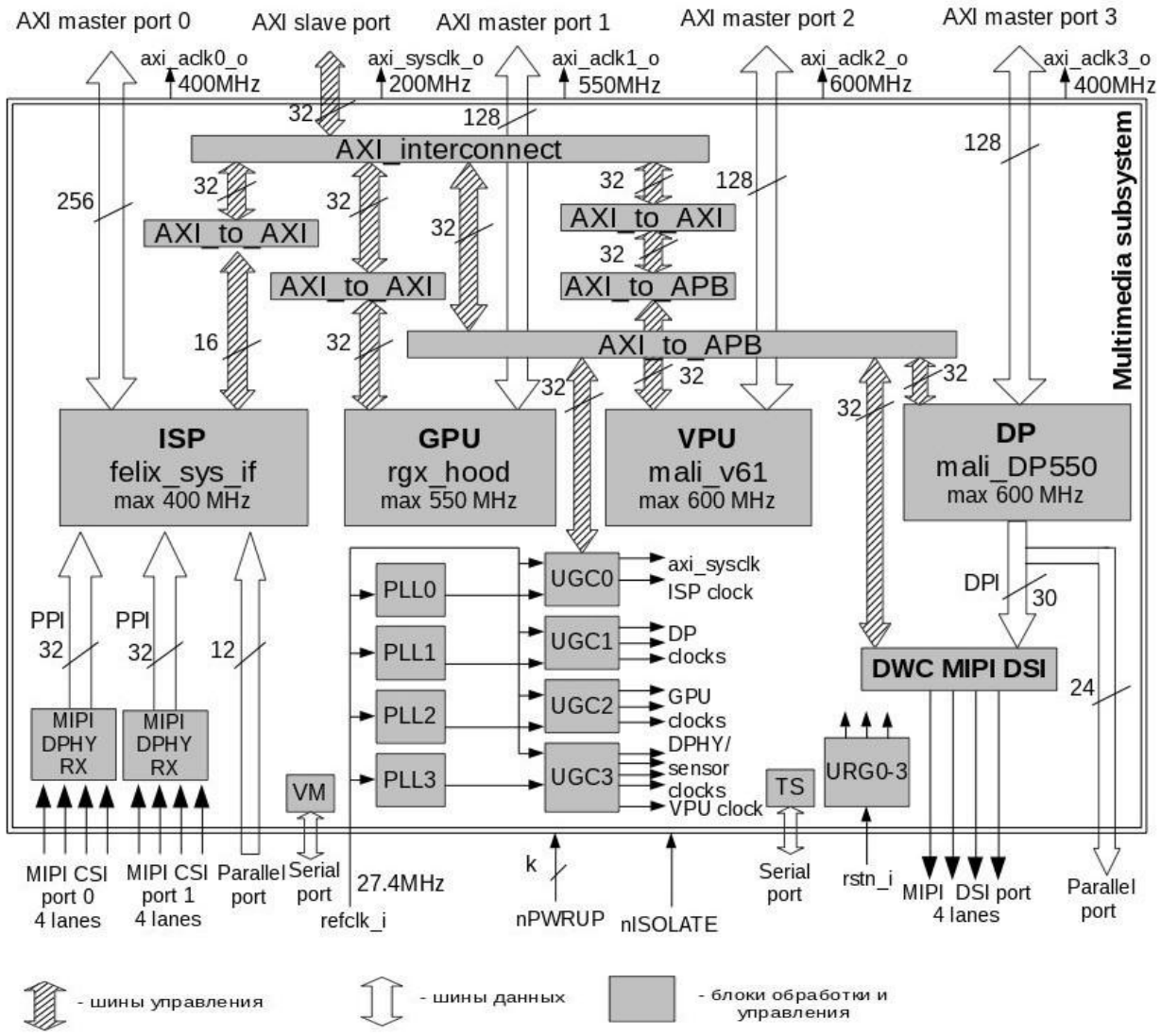


Рисунок 7.1. Структурная схема мультимедийной подсистемы

Перечень блоков управления и обработки, отображенных на структурной схеме, содержится в Таблица 7.1. Для крупных IP блоков (ISP, GPU, VPU, DP) указано наименование верхнего уровня иерархии блоков и максимальная частота. Количество переключателей питания nPWRUP k будет определено в процессе синтеза.

Таблица 7.1. Перечень блоков мультимедийной подсистемы

Наименование	Описание	IP блок
ISP	Блок предобработки видео (Image signal processor)	IMG PowerVR V2505 ISP Config 4
GPU	Графический процессор	IMG PowerVR SE8XE 4ppc GE8300 Clark
VPU	Видеокодер/декодер	ARM Mali V61 4c
DP	Процессор дисплея	ARM DP550
DWC MIPI DSI	Контроллер хоста MIPI DSI с встроенным передатчиком данных по стандарту MIPI	Synopys DWC MIPI DSI + Synopsys DWC MIPI DPHY Tx
MIPI DPHY RX	Приемник данных по стандарту MIPI	Synopsys DWC MIPI DPHY Rx
AXI interconnect	Коммутатор AXI шин	Synopsys DW_axi
AXI_to_AXI	Асинхронный мост между AXI шинами	Synopsys DW_axi_x2x
AXI_to_APB	Мост между AXI шиной и APB шиной	Synopsys DW_axi_x2p
PLL0-3	Блок PLL с контроллером	pll_ctr_tci28hpcp + True Circuit Clock-Generator PLL
UCG0-3	Универсальный блок формирования тактовой частоты	UCG
Reset controller	Блок управления сигналами сброса на основе универсального блока формирования сброса	URG + автоматы управления

7.2 Карта памяти

Адресное пространство мультимедийной подсистемы составляет 2 Мбайта. Карта памяти представлена в Таблица 7.2.

Таблица 7.2. Карта памяти мультимедийной подсистемы

Диапазон адресов	Описание	Размер адресного пространства
0x00_0000 - 0x07_FFFF	Регистры Graphics core Series8XE Clark	512 КБайт
0x08_0000 - 0x08_FFFF	Регистры Video Processor Mali-V61	64 КБайта
0x09_0000 - 0x09_FFFF	Резерв	
0x0A_0000 - 0x0A_FFFF	Регистры ISP Felix 2505	64 КБайта
0x0B_0000 - 0x0B_FFFF	Резерв	
0x0C_0000 - 0x0C_FFFF	Регистры контроллера DSI	64 КБайта
0x0D_0000 – 0x0F_FFFF	Резерв	
0x10_0000 - 0x10_FFFF	Регистры Display Processor Mali-DP550 secure/non-secure	64 КБайта
0x11_0000 - 0x11_FFFF	Регистры Display Processor Mali-DP550 secure	64 КБайта
0x12_0000 - 0x12_FFFF	Регистры управления/состояния подсистемы	64 КБайта
0x12_0000 - 0x12_0FFF	Регистры управления синхронизацией	4 КБайта
0x12_1000 - 0x12_1FFF	Регистры управления сбросом	4 КБайта

Диапазон адресов	Описание	Размер адресного пространства
0x12_2000 - 0x12_2FFF	Регистры конфигурации/управления подсистемой	4 КБайта
0x12_3000 - 0x12_3FFF	Регистры MIPI DPHY RX	4 КБайта
0x12_4000 - 0x12_4FFF	Регистры MIPI DPHY TX	4 КБайта
0x12_5000 - 0x12_5FFF	Регистры управления CMOS сенсорами	4 КБайта
0x12_6000 - 0x12_FFFF	Резерв	
0x13_0000 - 0x1F_FFFF	Резерв	

7.3 Регистры

Перечень регистров приведен в Таблица 7.3.

В перечне регистров используются следующие обозначения:

RW — read/write бит
RO — read/write бит
RR — read-reset бит (сбрасывается при чтении)
WO — write-only бит
W1C — write-1-clear (при записи единицы бит сбрасывается, запись 0 не оказывает влияния на состояние бита, при чтении возвращается текущее состояние бита)
W1SC — write-1-self-clear (записывается единица, сбрасывается аппаратурой)
U — значение после сброса не определено

Если особо не оговорено, из резервных полей регистров считываются нули. Запись в резервные поля регистров игнорируется. Запись регистров по резервным адресам игнорируется. При чтении регистров по резервным адресам возвращаются нули. При чтении WO полей возвращаются нули.

Доступ к регистрам осуществляется по адресам, выравненным на границу 32-разрядного слова. Два младших разряда адреса игнорируются.

Таблица 7.3. Перечень программно-доступных регистров

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
Регистры управления/состояния сигналов синхронизации (0x12_0000 – 0x12_0FFF)				

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
PLL0_CTRL	Регистр управления PLL0	0x00000000	RW	0x12_0000
PLL0_DIAG	Регистр диагностики PLL0	0x00000002	RO	0x12_0004
				0x12_0008-0x12_000C
PLL1_CTRL	Регистр управления PLL1	0x00000000	RW	0x12_0010
PLL1_DIAG	Регистр диагностики PLL1	0x00000002	RO	0x12_0014
-	Резерв			0x12_0018 - 0x12_001C
PLL2_CTRL	Регистр управления PLL2	0x00000000	RW	0x12_0020
PLL2_DIAG	Регистр диагностики PLL2	0x00000002	RO	0x12_0024
-	Резерв			0x12_0028 - 0x12_002C
PLL3_CTRL	Регистр управления PLL3	0x00000000	RW	0x12_0030
PLL3_DIAG	Регистр диагностики PLL3	0x00000002	RO	0x12_0034
-	Резерв			0x12_0038 - 0x12_003C
SYS_CLK_CTRL	Регистр управления синхросигналом шины управления	0x00000001	RW, W1SC	0x12_0040
ISP_CLK_CTRL	Регистр управления синхросигналом ISP Felix	0x00000001	RW, W1SC	0x12_0044
-	Резерв			0x12_0048-0x12_007C
UCG0_BP_CTR	Регистр управления переводом UCG0 в режим BYPASS	0x00000000	RW	0x12_0080
UCG0_SYNC_CLK	Регистр синхронизации каналов UCG0	0x00000000	WO	0x12_0084
-	Резерв			0x12_0088 – 0x12_00BC
DISPLAY_ACLK_CTRL	Регистр управления синхросигналом ACLK контроллера дисплея Mali DP550	0x00000001	RW, W1SC	0x12_00C0
DISPLAY_MCLK_CTRL	Регистр управления синхросигналом MCLK контроллера дисплея Mali DP550	0x00000001	RW, W1SC	0x12_00C4
DISPLAY_PXLCLK_CTRL	Регистр управления синхросигналом PXLCLK контроллера дисплея Mali DP550	0x00000001	RW, W1SC	0x12_00C8
-	Резерв			0x12_00CC-0x12_00FC
UCG1_BP_CTR	Регистр управления переводом UCG1 в режим BYPASS	0x00000000	RW	0x12_0100
UCG1_SYNC_CLK	Регистр синхронизации каналов UCG1	0x00000000	WO	0x12_0104
-	Резерв			0x12_0108 - 0x12_013C
GPU_SYS_CLK_CTRL	Регистр управления синхросигналом sys_clk GPU	0x00000002	RW, W1SC	0x12_0140
GPU_MEM_CLK_CTRL	Регистр управления синхросигналом mem_clk GPU	0x00000001	RW, W1SC	0x12_0144
GPU_CORE_CLK_CTRL	Регистр управления синхросигналом core_clk GPU	0x00000001	RW, W1SC	0x12_0148
-	Резерв			0x12_014C -

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
				0x12_017C
UCG2_BP_CTR	Регистр управления переводом UCG2 в режим BYPASS	0x00000000	RW	0x12_0180
UCG2_SYNC_CLK	Регистр синхронизации каналов UCG2	0x00000000	WO	0x12_0184
-	Резерв			0x12_0188-0x12_01BC
MIPI_RX_REFCLK_CTRL	Регистр управления синхросигналом refclk MIPI RX DPHY0-1	0x00000001	RW, W1SC	0x12_01C0
MIPI_RX0_CFG_CLK_CTRL	Регистр управления синхросигналом cfg_clk MIPI RX DPHY0	0x00000001	RW, W1SC	0x12_01C4
MIPI_RX1_CFG_CLK_CTRL	Регистр управления синхросигналом cfg_clk MIPI RX DPHY1	0x00000001	RW, W1SC	0x12_01C8
MIPI_TX_REFCLK_CTRL	Регистр управления синхросигналом refclk MIPI TX DPHY	0x00000001	RW, W1SC	0x12_01CC
MIPI_TX_CFG_CLK_CTRL	Регистр управления синхросигналом cfg_clk MIPI TX DPHY	0x00000001	RW, W1SC	0x12_01D0
CMOS0_CLK_CTRL	Регистр управления синхросигналом CMOS сенсора 0	0x00000001	RW, W1SC	0x12_01D4
CMOS1_CLK_CTRL	Регистр управления синхросигналом CMOS сенсора 1	0x00000001	RW, W1SC	0x12_01D8
MIPI_TXCLKESC_CLK_CTRL	Регистр управления синхросигналом txclkesc MIPI RX DPHY	0x00000001	RW	0x12_01DC
VPU_CLK_CTRL	Регистр управления синхросигналом видеопроцессора Mali-V61	0x00000001	RW, W1SC	0x12_01E0
-	Резерв			0x12_01E4 – 0x12_1FC
UCG3_BP_CTR	Регистр управления переводом UCG3 в режим BYPASS	0x00000000	RW	0x12_0200
UCG3_SYNC_CLK	Регистр синхронизации каналов UCG3	0x00000000	WO	0x12_0204
-	Резерв			0x12_0208-0x12_02EC
QCHANNEL_CTRL	Регистр управления QChannel интерфейсом	0x00000010	RW	0x12_02F0
-	Резерв			0x12_02F4 - 0x12_0FFC
Регистры управления/состояния сброса (0x12_1000 – 0x12_1FFC)				
ISP_PPOLICY	Регистр установки состояния ISP Felix	0x00000000	RW	0x12_1000
ISP_PSTATUS	Регистр отражения состояния ISP Felix	0x00000000	RO	0x12_1004
GPU_PPOLICY	Регистр установки состояния GPU	0x00000000	RW	0x12_1008
GPU_PSTATUS	Регистр отражения состояния	0x00000000	RO	0x12_100C

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
	GPU			
VPU_PPOLICY	Регистр установки состояния Mali V61	0x00000000	RW	0x12_1010
VPU_PSTATUS	Регистр отражения состояния Mali V61	0x00000000	RO	0x12_1014
DISPLAY_PPOLICY	Регистр установки состояния DP550	0x00000000	RW	0x12_1018
DISPLAY_PSTATUS	Регистр отражения состояния DP550	0x00000000	RO	0x12_101C
-	Резерв			0x12_1020-0x12_103C
UCG_RST_CTRL	Регистр управления сигналами сброса UCG	0x00000000	RW	0x12_1040
-	Резерв			0x12_1044-0x12_104C
MIPI_RX_RST_CTRL	Регистр управления сигналами сброса MIPI RX DPHY	0x00000000	RW	0x12_1050
CMOS_RST_CTRL	Регистр управления сигналами сброса CMOS сенсоров	0x00000000	RW	0x12_1054
-	Резерв			0x12_1058-0x12_1FFC
Регистры управления/состояния подсистемы (0x12_2000 – 0x12_2FFC)				
SUBSYSTEM_CFG	Регистр конфигурации подсистемы	0x00000000	RW	0x12_2000
SUBSYSTEM_CTRL	Регистр управления подсистемой	0x00000000	RW	0x12_2004
SUBSYSTEM_STATUS	Регистр состояния подсистемы	0x000100FF	RO,RR	0x12_2008
-	Резерв			0x12_200C-0x12_2FFC
Регистры управления/состояния MIPI RX DPHY (0x12_3000 – 0x12_3FFC)				
Регистры управления/состояния MIPI RX DPHY 0 (0x12_3000 – 0x12_303C)				
MIPI_RX0_CTRL	Регистр управления	0x00000000	RW	0x12_3000
MIPI_RX0_CLK_PARAM	Регистр параметров синхросигналов	0x00004929	W	0x12_3004
MIPI_RX0_STATUS	Регистр состояния	0x0042108F	RO	0x12_3008
	Резерв			0x12_300C
MIPI_RX0_TSTREG_CTRL	Регистр сигналов управления интерфейса с тестовыми регистрами	0x00000000	RW	0x12_3010
MIPI_RX0_TSTREG_DATA	Регистр сигналов данных интерфейса с тестовыми регистрами	0x00000000	RW, RO	0x12_3014
MIPI_RX0_TEST_CTRL	Регистр управления тестированием	0x00000000	RW	0x12_3018
MIPI_RX0_TEST_DATA	Регистр тестовых данных	0x00000000	RO	0x12_301C
-	Резерв			0x12_3020 - 0x12_303C
Регистры управления/состояния MIPI RX DPHY 1 (0x12_3040 – 0x12_3FFC)				
Регистры управления/состояния MIPI TX DPHY (0x12_4000 – 0x12_4FFC)				
MIPI_TX_CTRL	Регистр управления	0x00000000	RW	0x12_4000
MIPI_TX_CLK_PARAM	Регистр параметров синхросигналов	0x00004929	RW	0x12_4004

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
MIPI_TX_PLL_STATUS0	Регистр 0 состояния PLL	0x2000003F	RO	0x12_4008
MIPI_TX_PLL_STATUS1	Регистр 1 состояния PLL	0x00000000	RO	0x12_400C
MIPI_TX_TEST_CTRL	Регистр управления тестированием	0x00000000		0x12_4010
MIPI_TX_TEST_DATA	Регистр тестовых данных	0x00000000	RO	0x12_401C
-	Резерв			0x12_4020 – 0x12_4FFC
Регистры управления CMOS сенсоров (0x12_5000 – 0x12_5FFC)				
Регистры управления CMOS сенсора 0 (0x12_5000 – 0x12_500C)				
CMOS0_CTRL	Регистр управления	0x00000000	RW	0x12_5000
CMOS0_TIMER_HIGH	Регистр таймера кадровой синхронизации (высокий уровень)	0x00000000	RW	0x12_5004
CMOS0_TIMER_LOW	Регистр таймера кадровой синхронизации (низкий уровень)	0x00000000	RW	0x12_5008
-	Резерв			0x12_500C
Регистры управления CMOS сенсора 1 (0x12_5010- 0x12_501C)				
-	Резерв			0x12_5020 - 0x12_FFFC

7.3.1 Регистр PLL_CTRL0 (0x12_0000)

Регистр управления PLL0.

Формат регистра PLL_CTRL0 приведен в Таблица 7.4.

Таблица 7.4. Формат регистра PLL_CTRL0

Номер бита	Условное обозначение	Тип доступа	Назначение
7:0	SEL	RW	Значение коэффициента умножения частоты (1-128) 0 — bypass PLL 1 — Fref_clk *2 2 — Fref_clk *3 …. 115 — Fref_clk * 116 ... 255 — Fref_clk * 116 При SEL > 0 и MAN=1 конфигурация PLL определяется полями *_MAN.
8	-		Резерв
9	MAN	RW	1 — при SEL > 0 конфигурация PLL определяется полями *_MAN. Выходная частота опрелеляется по формуле Fref_clk * (RF_MAN+1) /(NF_MAN+1)/ (NR_MAN+1)
13:10	OD_MAN	RW	Коэффициент OD (валиден при PLL_MAN=1)
26:14	NF_MAN	RW	Коэффициент NF (валиден при PLL_MAN=1)
30:27	NR_MAN	RW	Коэффициент NR (валиден при PLL_MAN=1)
31	LOCK		Отображает состояние сигнала LOCK PLL0

7.3.2 Регистр PLL_DIAG0 (0x12_0004)

Регистр диагностики PLL0.

Формат регистра PLL_DIAG0 приведен в Таблица 7.5.

Таблица 7.5. Формат регистра PLL_DIAG0

Номер бита	Условное обозначение	Тип доступа	Назначение
0	TEST_DIAG	RW	1 — bypass опорной частоты на выход PLL
1	ENSAT_DIAG	RW	1 — разрешено насыщение
2	FASTEN_DIAG	RW	1 — разрешен fast lock
3	RFSLIP_DIAG	RO	Регистрирует ситуацию reference cycle sleep
4	FBSLIP_DIAG	RO	Регистрирует ситуацию feedback cycle sleep
31:5	-		Резерв

7.3.3 Регистр PLL_CTRL1 (0x12_0010)

Аналогично регистру управления PLL0.

7.3.4 Регистр PLL_DIAG1 (0x12_0014)

Аналогично регистру диагностики PLL0.

7.3.5 Регистр PLL_CTRL2 (0x12_0020)

Аналогично регистру управления PLL0.

7.3.6 Регистр PLL_DIAG2 (0x12_0024)

Аналогично регистру диагностики PLL0.

7.3.7 Регистр PLL_CTRL3 (0x12_0030)

Аналогично регистру управления PLL0.

7.3.8 Регистр PLL_DIAG3 (0x12_0034)

Аналогично регистру диагностики PLL0.

7.3.9 Регистр SYS_CLK_CTRL (0x12_0040)

Регистр управления синхросигналом шины управления.

Формат регистра SYS_CLK_CTRL приведен в Таблица 7.6. Формат регистра SYS_CLK_CTRL

Таблица 7.6. Формат регистра SYS_CLK_CTRL

Номер бита	Условное обозначение	Тип доступа	Назначение
0	LPI_EN	RW	Бит включения автоматического режима работы LowPower Должен быть равен 0 (ручной режим)

Номер бита	Условное обозначение	Тип доступа	Назначение
1	CLK_EN	RW	Бит управления частотой канала * в ручном режиме: 0 – частота выключена 1 – частота включена Значение после сброса: 1
9:2	-		Резерв
29:10	DIV_COEFF	RW	Коэффициент деления входной частоты Тип доступа: RW Значение по сбросу: DIVIDER_WIDTH'b0
30	DIV_LOCK	RO	Признак готовности делителя 0 - идет переключение на новый коэффициент деления 1 - переключение произведено Значение по сбросу: 1'b0
31	Резерв		

7.3.10 Регистр ISP_CLK_CTRL (0x12_0044)

Регистр управления синхросигнал ISP Felix. Формат регистра приведен в Таблица 7.7.

Таблица 7.7. Формат регистра ISP_CLK_CTRL

Номер бита	Условное обозначение	Тип доступа	Назначение
0	LPI_EN	RW	Бит включения автоматического режима работы LowPower интерфейса «Q-Channel» : 0 – автоматический режим выключен 1 – автоматический режим включен
1	CLK_EN	RW	Бит управления частотой канала в ручном режиме: 0 – частота выключена 1 – частота включена
4:2	CLK_EN_STS	RW	Отображение события, по которому был переключен бит CLK_EN CLK_EN_STATUS[0] = 1 - отключение частоты отклонено из-за занятости устройства (QDENY = 1) CLK_EN_STATUS[1] = 1 - выполнено отключение частоты по сигналу QACTIVE CLK_EN_STATUS[2] = 1 - выполнено включение частоты по сигналу QACTIVE Сброс статусных битов осуществляется записью "1" в соответствующий бит Значение по сбросу: 3'b0
5	-		Резерв
6	QACTIVE_CTL_EN	RW	Бит разрешения реакции на сигнал QACTIVE (разрешение на изменения состояния со стороны подключенного устройства) при работе в LPI (LPI_EN = 1) режиме: 0 – реакция запрещена 1 – реакция разрешена Бит ни на что не влияет, если LPI режим выключен (LPI_EN = 0) Значение по сбросу: 1'b0

Номер бита	Условное обозначение	Тип доступа	Назначение
9:7	FSM_STATE	RO	Биты отображения состояния LowPower FSM: 000 – Q_STOPPED 010 – Q_REQUEST 110 – Q_RUN 100 – Q_EXIT 011 – Q_DENIED 001 – Q_CLK_EN 111 – Q_CONTINUE
29:10	DIV_COEFF	RW	Коэффициент деления входной частоты 0 – деление на 1 1 – деление на 1 2 – деление на 2 3 – деление на 3 … 2^{20}-1 – деление на 2^{20}-1
30	DIV_LOCK	RO	Признак готовности делителя 0 - идет переключение на новый коэффициент деления 1 - переключение произведено Значение по сбросу: 1’b0
31	Резерв		

7.3.11 Регистр UCG0_BP_CTRL (0x12_0080)

Регистр управления BYPASS mode UCG0. Формат регистра приведен в Таблица 7.8.

Таблица 7.8. Формат регистра UCG0_BP_CTR

Номер бита	Условное обозначение	Тип доступа	Назначение
0	XTI_EN_CH0	RW	Бит включения BYPASS mode канала 0 : 0 – mission mode 1 – bypass mode
1	XTI_EN_CH1	RW	Бит включения BYPASS mode канала 1 : 0 – mission mode 1 – bypass mode
31:2	-		Резерв

7.3.12 Регистр UCG0_SYNC_CLK (0x12_0084)

Регистр управления синхронизации каналов UCG0. Формат регистра приведен в Таблица 7.9.

Таблица 7.9. Формат регистра UCG0_SYNC_CLK

Номер бита	Условное обозначение	Тип доступа	Назначение
0	SYNC_CLK_CH0	WO	Запрос на синхронизацию канала 0. Активный уровень «1»

Номер бита	Условное обозначение	Тип доступа	Назначение
1	SYNC_CLK_CH1	WO	Запрос на синхронизацию канала 1. Активный уровень «1»
31:2	-		Резерв

7.3.13 Регистр DISPLAY_ACLK_CTRL (0x12_00C0)

Регистр управления синхросигналом ACLK контроллера дисплея Mali DP550

Аналогично ISP_CLK_CTRL.

7.3.14 Регистры DISPLAY_MCLK_CTRL-DISPLAY_PXLCLK_CTRL (0x12_00C4 - 0x12_0C8)

Регистр управления синхросигналами MCLK-PXLCLK контроллера дисплея Mali DP550

Аналогично SYS_CLK_CTRL.

7.3.15 Регистр UCG1_BP_CTRL (0x12_0100)

Регистр управления BYPASS mode UCG1. Формат регистра приведен в Таблица 7.10.

Таблица 7.10. Формат регистра UCG1_BP_CTR

Номер бита	Условное обозначение	Тип доступа	Назначение
0	XTI_EN_CH0	RW	Бит включения BYPASS mode канала 0 : 0 – mission mode 1 – bypass mode
1	XTI_EN_CH1	RW	Бит включения BYPASS mode канала 1 : 0 – mission mode 1 – bypass mode
2	XTI_EN_CH2	RW	Бит включения BYPASS mode канала 2 : 0 – mission mode 1 – bypass mode
31:3	-		Резерв

7.3.16 Регистр UCG1_SYNC_CLK (0x12_0104)

Регистр управления синхронизации каналов UCG1. Формат регистра приведен в Таблица 7.11.

Таблица 7.11. Формат регистра UCG1_SYNC_CLK

Номер бита	Условное обозначение	Тип доступа	Назначение
0	SYNC_CLK_CH0	WO	Запрос на синхронизацию канала 0. Активный уровень «1»
1	SYNC_CLK_CH1	WO	Запрос на синхронизацию канала 1. Активный уровень «1»
2	SYNC_CLK_CH2	WO	Запрос на синхронизацию канала 1. Активный уровень «1»

Номер бита	Условное обозначение	Тип доступа	Назначение
31:3	-		Резерв

7.3.17 Регистры GPU_*_CLK_CTRL (0x12_0140 - 0x12_0148)

Регистры управления синхросигналами GPU.

Аналогично SYS_CLK_CTRL.

7.3.18 Регистр UCG2_BP_CTRL (0x12_0180)

Регистр управления BYPASS mode UCG2. Аналогично регистру управления BYPASS mode UCG1.

7.3.19 Регистр UCG2_SYNC_CLK (0x12_0184)

Регистр управления синхронизацией каналов UCG2. Аналогично регистру управления BYPASS mode UCG1.

7.3.20 Регистры MIPI/CMOS*_CLK_CTRL (0x12_01C0 - 0x12_01DC)

Регистры управления синхросигналами MIPI DPHY и CMOS сенсоров.

Аналогично SYS_CLK_CTRL.

7.3.21 Регистр VPU_CLK_CTRL (0x12_01E0)

Регистр управления синхросигналом видеопроцессора Mali-V61.

Аналогично ISP_CLK_CTRL.

7.3.22 Регистр управления BYPASS mode UCG3 (0x12_0200)

Регистр управления BYPASS mode UCG3. Формат регистра приведен в Таблица 7.12.

Таблица 7.12. Формат регистра UCG3_BP_CTR

Номер бита	Условное обозначение	Тип доступа	Назначение
0	XTI_EN_CH0	RW	Бит включения BYPASS mode канала 0 : 0 – mission mode 1 – bypass mode
1	XTI_EN_CH1	RW	Бит включения BYPASS mode канала 1 : 0 – mission mode 1 – bypass mode
2	XTI_EN_CH2	RW	Бит включения BYPASS mode канала 2 : 0 – mission mode 1 – bypass mode

Номер бита	Условное обозначение	Тип доступа	Назначение
3	XTI_EN_CH3	RW	Бит включения BYPASS mode канала 3 : 0 – mission mode 1 – bypass mode
4	XTI_EN_CH4	RW	Бит включения BYPASS mode канала 4 : 0 – mission mode 1 – bypass mode
5	XTI_EN_CH5	RW	Бит включения BYPASS mode канала 5 : 0 – mission mode 1 – bypass mode
6	XTI_EN_CH6	RW	Бит включения BYPASS mode канала 6 : 0 – mission mode 1 – bypass mode
7	XTI_EN_CH7	RW	Бит включения BYPASS mode канала 2: 0 – mission mode 1 – bypass mode
8	XTI_EN_CH8	RW	Бит включения BYPASS mode канала 8 : 0 – mission mode 1 – bypass mode
31:9	-		Резерв

7.3.23 Регистр синхронизации каналов UCG3 (0x12_0204)

Регистр управления BYPASS mode UCG3. Формат регистра приведен в Таблица 7.13.

Таблица 7.13. Формат регистра UCG3_SYNC_CLK

Номер бита	Условное обозначение	Тип доступа	Назначение
0	SYNC_CLK_CH0	WO	Запрос на синхронизацию канала 0. Активный уровень «1»
1	SYNC_CLK_CH1	WO	Запрос на синхронизацию канала 1. Активный уровень «1»
2	SYNC_CLK_CH2	WO	Запрос на синхронизацию канала 2. Активный уровень «1»
3	SYNC_CLK_CH3	WO	Запрос на синхронизацию канала 3. Активный уровень «1»
4	SYNC_CLK_CH4	WO	Запрос на синхронизацию канала 4. Активный уровень «1»
5	SYNC_CLK_CH5	WO	Запрос на синхронизацию канала 5. Активный уровень «1»
6	SYNC_CLK_CH6	WO	Запрос на синхронизацию канала 6. Активный уровень «1»
7	SYNC_CLK_CH7	WO	Запрос на синхронизацию канала 7. Активный уровень «1»
8	SYNC_CLK_CH8	WO	Запрос на синхронизацию канала 8. Активный уровень «1»
31:9	-		Резерв

7.3.24 Регистр QCHANNEL_CTRL (0x12_02F0)

Регистр управления интерфейсом отключения сигналов синхронизации.

Формат регистра QCHANNEL_CTRL приведен в Таблица 7.14.

Таблица 7.14. Формат регистра QCHANNEL_CTRL

Номер бит	Условное обозначение	Тип доступа	Назначение
7:0	ISP_IDLE_TIMEOUT	RW	Время ожидания установки сигнала isp_idle при запросе на отключение синхросигнала ISP в тактах синхросигнала (1-255)
31:8	-		Резерв

7.3.25 Регистр ISP_PPOLICY (0x12_1000)

Регистр установки состояния ISP Felix.

Формат регистра ISP_PPOLICY приведен в Таблица 7.15.

Таблица 7.15. Формат регистра ISP_PPOLICY

Номер бита	Условное обозначение	Тип доступа	Назначение
4:0	POLICY	RW	Установка состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; другое - резерв
31:5	-		Резерв

7.3.26 Регистр ISP_PSTATUS (0x12_1004)

Регистр отражения состояния ISP Felix.

Формат регистра ISP_PSTATUS приведен в Таблица 7.16.

Таблица 7.16. Формат регистра ISP_PSTATUS

Номер бита	Условное обозначение	Тип доступа	Назначение
4:0	PSTATUS	RO	Поле отражает текущую активную установку состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; другое - резерв/выключено
31:5	-		Резерв

7.3.27 Регистры GPU_PPOLICY/GPU_PSTATUS (0x12_1008-0x12_100C)

Регистры установки/отражения состояния GPU.

Аналогично регистрам установки/отражения состояния ISP Felix ISP_PPOLICY/ISP_PSTATUS.

7.3.28 Регистры VPU_PPOLICY/VPU_PSTATUS (0x12_1010-0x12_1014)

Регистры установки/отражения состояния Mali V61.

Аналогично регистрам установки/отражения состояния ISP Felix ISP_PPOLICY/ISP_PSTATUS.

7.3.29 Регистры DISPLAY_PPOLICY/DISPLAY_PSTATUS (0x12_1018-0x12_101C)

Регистры установки/отражения состояния DP 550.

Аналогично регистрам установки/отражения состояния ISP Felix ISP_PPOLICY/ISP_PSTATUS.

7.3.30 Регистр UCG_RST_CTRL (0x12_1040)

Регистр управления сигналами сброса UCG0-3.

Формат регистра UCG_RST_CTRL приведен в Таблица 7.17.

Таблица 7.17. Формат регистра UCG_RST_CTRL

Номер бита	Условное обозначение	Тип доступа	Назначение
0	UCG0_RST	RW	Управляет сигналом сброса UCG 0
1	UCG1_RST	RW	Управляет сигналом сброса UCG 1
2	UCG2_RST	RW	Управляет сигналом сброса UCG 2
3	UCG3_RST	RW	Управляет сигналом сброса UCG 3
31:4	-		Резерв

7.3.31 Регистр MIPI_RX_RST_CTRL (0x12_1050)

Регистр управления сигналами сброса MIPI DPHY0-1.

Формат регистра MIPI_RX_RST_CTRL приведен в Таблица 7.18 .

Таблица 7.18. Формат регистра UCG_RST_CTRL

Номер бита	Условное обозначение	Тип доступа	Назначение
0	MIPI_RX0_RSTZ	RW	Управляет сигналом сброса rstz MIPI RX DPHY0 0 – сброс активен
1	MIPI_RX0_SHUTSOWNZ	RW	Управляет сигналом сброса shutdownz MIPI RX DPHY0 0 – shutdown
15:2	-		Резерв
16	MIPI_RX1_RSTZ	RW	Управляет сигналом сброса rstz MIPI RX DPHY1 0 – сброс активен
17	MIPI_RX1_SHUTSOWNZ	RW	Управляет сигналом сброса shutdownz MIPI RX DPHY1 0 – shutdown
31:4	-		Резерв

7.3.32 Регистр CMOS_RST_CTRL (0x12_1054)

Регистр управления сигналами сброса блоков управления CMOS сенсорами.

Формат регистра UCG_RST_CTRL приведен в Таблица 7.19.

Таблица 7.19. Формат регистра UCG_RST_CTRL

Номер бита	Условное обозначение	Тип доступа	Назначение
0	CMOS0_RST	RW	Управляет сигналом сброса блока управления CMOS сенсором 0
1	-		Резерв
2	CMOS1_RST	RW	Управляет сигналом сброса блока управления CMOS ом 1
31:4	-		Резерв

7.3.33 Регистр SUBSYSTEM_CFG (0x12_2000)

Регистр конфигурации подсистемы.

Формат регистра SUBSYSTEM_CFG приведен в Таблица 7.20.

Таблица 7.20. Формат регистра SUBSYSTEM_CFG

Номер бита	Условное обозначение	Тип доступа	Назначение
0	DISPLAY_PARALLEL_PORT_EN	RW	1 — включен параллельный порт дисплея
4:1	-		Резерв
5:4	TEST_DATA_SRC		0x – на шину VTEST выдаются нули 10 – на шину VTEST выдаются сигналы ISP rtm_a 11 – на шину VTEST выдаются сигналы ISP rtm_b
7:6	-		Резерв
8	VPU_DISABLE_AFBC	RW	1 — запрещена AFBC компрессия
9	VPU_DISABLE_REAL	RW	1 — запрещено RealVideo
10	VPU_DISABLE_VPX	RW	1 — запрещены VP8 и VP9
11	VPU_DISABLE_HEVC	RW	1 — запрещен HVEC (H.265)
15:12	-		Резерв
16	GPU_ENABLE_DXT	RW	1 — разрешена dxt компрессия текстур
17	GPU_ENABLE_ASTC	RW	1 — разрешена Adaptive Scalable Texture Compression
23:16	-		
24	DSI_DPI_DATA_BUS_WIDTH	RW	0 — 24-битный режим (8 бит на цвет) 1 — 30-битный режим (10 бит на цвет)
31:25	-		Резерв

7.3.34 Регистр SUBSYSTEM_CTRL (0x12_2004)

Регистр управления подсистемой.

Формат регистра SUBSYSTEM_CTRL приведен в Таблица 7.21.

Таблица 7.21. Формат регистра SUBSYSTEM_CTRL

Номер бита	Условное обозначение	Тип доступа	Назначение
3:0	-		Резерв
4	ISP_MEM_LIMITED_THROUGHPUT	RW	1 — ограничиваются низкоприоритетные запросы на пересылку данных от ISP
8:5	-		Резерв
9	DSI_SHUTDOWN	RW	1 — инициирует выдачу команды на shutdown

			монитора
10	DSI_COLORMODE	RW	0 — нормальный цветовой режим 1 — режим low color
31:10	-		Резерв

7.3.35 Регистр SUBSYSTEM_STATUS (0x12_2008)

Регистр состояния подсистемы.

Формат регистра SUBSYSTEM_STATUS приведен в Таблица 7.22.

Таблица 7.22. Формат регистра SUBSYSTEM_STATUS

Номер бита	Условное обозначение	Тип доступа	Назначение
0	DISPLAY_V1_PL1_FIFO_LEVEL	RO	Количество слов в FIFO для слоя видео1 (плэйн 1): 0 — больше или равно THRESHOLD_HIGH 1 — меньше THRESHOLD_LOW
1	DISPLAY_V1_PL2_FIFO_LEVEL	RO	Количество слов в FIFO для слоя видео1 (плэйн 2): 0 — больше или равно THRESHOLD_HIGH 1 — меньше THRESHOLD_LOW
2	DISPLAY_V1_PL3_FIFO_LEVEL	RO	Количество слов в FIFO для слоя видео1 (плэйн 3): 0 — больше или равно THRESHOLD_HIGH 1 — меньше THRESHOLD_LOW
3	DISPLAY_V2_PL1_FIFO_LEVEL	RO	Количество слов в FIFO для слоя видео2 (плэйн 1): 0 — больше или равно THRESHOLD_HIGH 1 — меньше THRESHOLD_LOW
4	DISPLAY_V2_PL2_FIFO_LEVEL	RO	Количество слов в FIFO для слоя видео2 (плэйн 2): 0 — больше или равно THRESHOLD_HIGH 1 — меньше THRESHOLD_LOW
5	DISPLAY_V2_PL3_FIFO_LEVEL	RO	Количество слов в FIFO для слоя видео2 (плэйн 3): 0 — больше или равно THRESHOLD_HIGH 1 — меньше THRESHOLD_LOW
6	DISPLAY_GRAPH_FIFO_LEVEL	RO	Количество слов в FIFO для слоя графики: 0 — больше или равно THRESHOLD_HIGH 1 — меньше THRESHOLD_LOW
7	DISPLAY_SMART_FIFO_LEVEL	RO	Количество слов в FIFO для слоя smart: 0 — больше или равно THRESHOLD_HIGH 1 — меньше THRESHOLD_LOW
8	DISPLAY_MMUPROT	RO	1 — дисплей в protected mode
9	DISPLAY_PREFETCH_START	RO	1 — дисплей выполняет prefetch
15:10	-		Резерв
16	DSI_DPI_HALT	RO	0 — данные передаются по DPI интерфейсу 1 — DPI интерфейс остановлен
19:17	-		Резерв
20	VPU_SLAVE_ERROR	RR	1 — VPU выдал ошибку SLAVE ERROR по APB
31:21	-		Резерв

7.3.36 Регистр MIPI_RX0_CTRL (0x12_300)

Регистр управления приемником видеосигнала MIPI DPHY RX. Формат регистра MIPI DPHY RX приведен в Таблица 7.23.

Таблица 7.23. Формат регистра MIPI_RX0_CTRL

Номер бита	Условное обозначение	Тип доступа	Назначение
0	PHY_ENABLECLK	RW	1 — работа clock lane разрешена
1	PHY_ENABLE_RX_LANE0	RW	1 — работа data lane 0 разрешена
2	PHY_ENABLE_RX_LANE1	RW	1 — работа data lane 1 разрешена
3	PHY_ENABLE_RX_LANE2	RW	1 — работа data lane 2 разрешена
4	PHY_ENABLE_RX_LANE3	RW	1 — работа data lane 3 разрешена
5	PHY_TURN_DISABLE0	RW	1 — запрос turn around для lane 0 запрещен
8:6	-		Резерв
9	PHY_FORCE_RXMODE0	RW	1 — data lane 0 устанавливается в режим приема
10	PHY_FORCE_RXMODE1	RW	1 — data lane 1 устанавливается в режим приема
11	PHY_FORCE_RXMODE2	RW	1 — data lane 2 устанавливается в режим приема
12	PHY_FORCE_RXMODE3	RW	1 — data lane 3 устанавливается в режим приема
13	PHY_TURN_REQUEST0	RW	1 — запрос turn around для lane 0
16:14	-		Резерв
17	PHY_FORCETXSTOPMODE0	RW	1 — data lane 0 переходит в режим передачи и его SM устанавливается в stop state
20:18	-		Резерв
21	SHADOW_CLEAR	RW	Сброс теневых регистров (импульс)
31:22	-		Резерв

7.3.37 Регистр MIPI_RX0_CLK_PARAM (0x12_3004)

Регистр параметров синхросигналов приемника видеосигнала MIPI DPHY RX. Формат регистра MIPI_RX0_CLK_PARAM приведен в Таблица 7.24.

Таблица 7.24. Формат регистра MIPI_RX0_CLK_PARAM

Номер бита	Условное обозначение	Тип доступа	Назначение
7:0	CFG_CLK_FREQ_RANGE	RW	Предустановленное значение частоты клока конфигурации: round[(Fcfg_clk(MHZ)-17)*4]
14:8	HS_FREQ_RANGE	RW	Диапазон операционных частот. Устанавливается во время powerdown и не должен меняться во время работы
31:15	-		Резерв

7.3.38 Регистр MIPI_RX0_STATUS (0x12_3008)

Регистр состояния приемника видеосигнала MIPI DPHY RX. Формат регистра MIPI_RX0_STATUS приведен в Таблица 7.25.

Таблица 7.25. Формат регистра MIPI_RX0_STATUS

Номер бита	Условное обозначение	Тип доступа	Назначение
0	STOP_STATE_CLK	RO	1 — clock лэйн в состоянии останова
1	ULPS_CLK_N	RO	0 — clock лэйн 0 вошел в состояние Ultra Low Power, сохраняется активным пока clock лэйн в состоянии останова
2	ULPS_CLK_ACTIVE_N	RO	0 — лэйн 0 в состоянии Ultra Low Power
3	CLK_ACTIVE_HS	RO	1 — clock lane в состоянии high speed
4	DIRECTION_0	RO	Текущее направление лэйна 0:

Номер бита	Условное обозначение	Тип доступа	Назначение
			0 — передатчик, 1 — приемник
5	ERRCONTROL_0	RO	1 — обнаружена некорректная последовательность состояний лэйна 0
6	ERRCONTENTIONLP_0_0	RO	1 — лэйн 0 с базовым направлением RX функционирует как TX и выдает ноль
7	ERRCONTENTIONLP_1_0	RO	1 — лэйн 0 с базовым направлением RX функционирует как TX и выдает единицу
8	ULPS_ACTIVE_N_0	RO	0 — лэйн 0 в состоянии Ultra Low Power
9	ULPS_ESC_0	RO	0 — лэйн 0 в состоянии Ultra Low Power и Stop state
10	STOP_STATE_0	RO	1 — лэйн 0 в Stop State
11	ERRCONTROL_1	RO	1 — обнаружена некорректная последовательность состояний лэйна 1
12	ULPS_ACTIVE_N_1	RO	0 — лэйн 1 в состоянии Ultra Low Power
13	ULPS_ESC_1	RO	0 — лэйн 1 в состоянии Ultra Low Power и Stop state
14	ERRCONTROL_2	RO	1 — обнаружена некорректная последовательность состояний лэйна 2
15	STOP_STATE_1	RO	1 — лэйн 1 в Stop State
16	ULPS_ACTIVE_N_2	RO	0 — лэйн 2 в состоянии Ultra Low Power
17	ULPS_ESC_2	RO	0 — лэйн 2 в состоянии Ultra Low Power и Stop state
18	STOP_STATE_2	RO	1 — лэйн 2 в Stop State
19	ERRCONTROL_3	RO	1 — обнаружена некорректная последовательность состояний лэйна 3
20	ULPS_ACTIVE_N_3	RO	0 — лэйн 3 в состоянии Ultra Low Power
21	ULPS_ESC_3	RO	0 — лэйн 3 в состоянии Ultra Low Power и Stop state
22	STOP_STATE_3	RO	1 — лэйн 3 в Stop State
23	SKEW_CAL_HS	RO	1 — DPHY вошел в процесс deskew калибровки
31:24	-		Резерв

7.3.39 Регистр MIPI_RX0_TSTREG_CTRL (0x12_3010)

Регистр сигналов управления интерфейса с тестовыми регистрами приемника видеосигнала MIPI DPHY RX. Формат регистра MIPI_RX0_TSTREG_CTRL приведен в Таблица 7.29.

Таблица 7.26. Формат регистра MIPI_RX0_TSTREG_CTRL

Номер бита	Условное обозначение	Тип доступа	Назначение
0	TESTCLR	RW	1 — сброс тестового интерфейса активен
1	TESTCLK	RW	Бит используется как синхросигнал тестового интерфейса
31:2	-		Резерв

7.3.40 Регистр MIPI_RX0_TSTREG_DATA (0x12_3014)

Регистр сигналов данных интерфейса с тестовыми регистрами приемника видеосигнала MIPI DPHY RX. Формат регистра MIPI_RX0_TSTREG_DATA приведен в Таблица 7.27.

Таблица 7.27. Формат регистра MIPI_RX0_TSTREG_DATA

Номер бита	Условное обозначение	Тип доступа	Назначение
7:0	TESTDIN	RW	Шина данных для программирования тесовых регистров
15:8	TESTDOUT	RO	Шина данных для чтения тестовых регистров

Номер бита	Условное обозначение	Тип доступа	Назначение
16	TESTEN	RW	Селектор операции: 1 — операция записи данных 0 — операция записи адреса
31:17	-		Резерв

7.3.41 Регистр MIPI_RX0_TEST_CTRL (0x12_3018)

Регистр управления тестированием приемника видеосигнала MIPI DPHY RX. Формат регистра MIPI_RX0_TEST_CTRL приведен в Таблица 7.28.

Таблица 7.28. Формат регистра MIPI_RX0_TEST_CTRL

Номер бита	Условное обозначение	Тип доступа	Назначение
0	BIST_ON	RW	1 — стартует BIST тест
1	BIST_DONE	RO	1 — BIST тест завершен
2	BIST_OK	RO	1 — BIST тест завершен успешно
3	Резерв		
4	CONT_EN	RW	1 — DPHY переводится в режим непрерывного тестирования. Состояние лэйнов отображается на шине cont_data
7:5	-		Резерв
9:8	PLL_CLK_SEL	RW	Управление выходом PLL: 00 — клок остановлен 01 — клок генерируется 10 — на выход выдается буферизованный референсный клок 11 — запрещенная комбинация
31:10	-		Резерв

7.3.42 Регистр MIPI_RX0_TEST_DATA (0x12_301C)

Регистр тестирования приемника видеосигнала MIPI DPHY RX. Формат регистра MIPI_RX0_TEST_DATA приведен в Таблица 7.29.

Таблица 7.29. Формат регистра MIPI_RX0_TEST_DATA

Номер бита	Условное обозначение	Тип доступа	Назначение
10:0	CONT_DATA	RO	Сигналы непрерывного тестирования
31:11	Резерв		

7.3.43 Регистры управления/состояния MIPI RX DPHY 1 (0x12_3040 – 0x12_3058)

Аналогично регистрам управления/состояния MIPI RX DPHY0.

7.3.44 Регистр MIPI_TX_CTRL (0x12_4000)

Регистр управления передатчиком видеосигнала MIPI DPHY TX. Формат регистра MIPI_TX_CTRL приведен в Таблица 7.30.

Таблица 7.30. Формат регистра MIPI_TX_CTRL

Номер бита	Условное обозначение	Тип доступа	Назначение
4:0	-		Резерв
5	PHY_TURN_DISABLE0	RW	1 — запрос turn around для lane 0 запрещен
8:6	-		Резерв
9	PHY_FORCE_RXMODE0	RW	1 — data lane 0 устанавливается в режим приема
16:10	-		Резерв
17	PHY_FORCETXSTOPMODE0	RW	1— data lane 0 переходит в режим передачи и его SM устанавливается в stop state
18	PHY_FORCETXSTOPMODE1	RW	1— data lane 1 переходит в режим передачи и его SM устанавливается в stop state
19	PHY_FORCETXSTOPMODE2	RW	1— data lane 2 переходит в режим передачи и его SM устанавливается в stop state
20	PHY_FORCETXSTOPMODE3	RW	1— data lane 3 переходит в режим передачи и его SM устанавливается в stop state
21	SHADOW_CLEAR	RW	Сброс теневых регистров (импульс)
23:22	-		Резерв
24	PLL_FORCE_LOCK	RW	1— принудительно устанавливается lock
26:25	PLL_CLK_SEL	RW	Управление выходом PLL: 00 — клок остановлен 01 — клок генерируется 10 — на выход выдается буферизованный референсный клок 11 — запрещенная комбинация
31:27	-		Резерв

7.3.45 Регистр MIPI_TX_CLK_PARAM (0x12_4004)

Регистр параметров синхросигналов передатчика видеосигнала MIPI DPHY TX. Формат регистра MIPI_TX_CLK_PARAM приведен в Таблица 7.31.

Таблица 7.31. Формат регистра MIPI_TX_CLK_PARAM

Номер бита	Условное обозначение	Тип доступа	Назначение
7:0	CFG_CLK_FREQ_RANGE	RW	Предустановленное значение частоты клока конфигурации: round[(Fcfg_clk(MHZ)-17)*4]
14:8	HS_FREQ_RANGE	RW	Диапазон операционных частот. Устанавливается во время powerdown и не должен меняться во время работы
31:15	-		Резерв

7.3.46 Регистр MIPI_TX_PLL_STATUS0 (0x12_4008)

Регистр состояния 0 передатчика видеосигнала MIPI DPHY TX. Формат регистра MIPI_TX_STATUS0 приведен в Таблица 7.32.

Таблица 7.32. Формат регистра MIPI_TX_STATUS0

Номер бита	Условное обозначение	Тип доступа	Назначение
5:0	VCO_CNTRL_OBS	RO	Состояние сигнала управления VCO PLL

Номер бита	Условное обозначение	Тип доступа	Назначение
11:6	PROP_CNTRL_OBS	RO	Состояние сигнала управления пропорциональным Charge Pump PLL
13:12	-		Резерв
19:14	INT_CNTRL_OBS	RO	Состояние сигнала управления интегральным Charge Pump PLL
21:20	GMP_CTRLS_OBS	RO	Состояние сигнала управления GMP PLL
28:22	CPBIAS_CTRLS_OBS	RO	Состояние сигнала управления Charge Pump Bias PLL
29	PLL_SHADOW_CTRLS_OBS	RO	Состояние сигнала механизма конфигурации PLL: 0 — со стороны SOC 1 — через DPHY
31:30	-		Резерв

7.3.47 Регистр MIPI_TX_PLL_STATUS1 (0x12_400C)

Регистр состояния 1 передатчика видеосигнала MIPI DPHY TX. Формат регистра MIPI_TX_STATUS1 приведен в Таблица 7.33.

Таблица 7.33. Формат регистра MIPI_TX_STATUS1

Номер бита	Условное обозначение	Тип доступа	Назначение
9:0	M_CNTRL_OBS	RO	Состояние сигнала управления m PLL
13:10	N_CNTRL_OBS	RO	Состояние сигнала управления n PLL
31:14	-		Резерв

7.3.48 Регистр MIPI_TX_TEST_CTRL (0x12_4010)

Регистр управления тестированием передатчика видеосигнала MIPI DPHY TX. Формат регистра MIPI_TX_TEST_CTRL приведен в Таблица 7.41.

Таблица 7.34. Формат регистра MIPI_TX_TEST_CTRL

Номер бита	Условное обозначение	Тип доступа	Назначение
0	BIST_ON	RW	1 — стартует BIST тест
3:1	-		Резерв
4	CONT_OK	RW	1 — DPHY переводится в режим непрерывного тестирования. Состояние лэйнов отображается на шине cont_data
31:5	-		Резерв

7.3.49 Регистр MIPI_TX_TEST_DATA (0x12_4014)

Регистр тестовых данных передатчика видеосигнала MIPI DPHY TX. Формат регистра MIPI_TX_TEST_DATA приведен в Таблица 7.35.

Таблица 7.35. Формат регистра MIPI_TX_TEST_DATA

Номер бита	Условное обозначение	Тип доступа	Назначение
10:0	CONT_DATA	RO	Сигналы непрерывного тестирования
31:11	Резерв		

7.3.50 Регистр CMOS0_CTRL (0x12_5000)

Регистр управления CMOS сенсором 0. Формат регистра CMOS0_CTRL приведен в Таблица 7.36.

Таблица 7.36. Формат регистра CMOS0_CTRL

Номер бита	Условное обозначение	Тип доступа	Назначение
0	FSYNC_ENABLE	RW	1 — выдача сигнала FSYNC разрешена
1	FSYNC_ENABLE_EDGE	RW	Уровень сигнала FSYNCO_0, устанавливаемый по положительному фронту FSYNC_ENABLE
31:14	-		Резерв

7.3.51 Регистр CMOS0_TIMER_HIGH (0x12_5004)

Регистр таймера кадровой синхронизации (высокий уровень) CMOS сенсора 0. Формат регистра CMOS0_TIMER_HIGH приведен в Таблица 7.37.

Таблица 7.37. Формат регистра CMOS0_TIMER_HIGH

Номер бита	Условное обозначение	Тип доступа	Назначение
31:0	TIMER_HIGH	RW	Длительность импульса уровня логической единицы сигнала кадровой синхронизации FSYNCO_0 (в тактах PCLKO_0)

7.3.52 Регистр CMOS0_TIMER_LOW (0x12_5008)

Регистр таймера кадровой синхронизации (низкий уровень) CMOS сенсора 0. Формат регистра CMOS0_TIMER_LOW приведен в Таблица 7.38.

Таблица 7.38. Формат регистра CMOS0_TIMER_LOW

Номер бита	Условное Обозначение	Тип доступа	Назначение
31:0	TIMER_LOW	RW	Длительность импульса уровня логического нуля сигнала кадровой синхронизации FSYNCO_0 (в тактах PCLKO_0)

7.3.53 Регистры управления CMOS сенсора 1 (0x12_5010 - 0x12_5018)

Аналогично регистрам управления CMOS сенсора 1.

7.4 Функционирование

7.4.1 Инициализация

Вся логика мультимедийной подсистемы находится в одном отключаемом домене питания MEDIA. Включение и выключение домена осуществляется системным контроллером PMCTR. После включения питания микросхемы домен MEDIA находится в выключенном состоянии. Для включения домена PMCTR осуществляет следующую последовательность действий:

1. Включает переключатели питания nPWRUP.
2. Снимает изоляцию установкой в состояние логической единицы сигнала nISOLATE.
3. Устанавливает в неактивное состояние (логической единицы) сигнал входного асинхронного сброса подсистемы resetn_i.
4. Включает сигнал входной референсной частоты refclk_i.

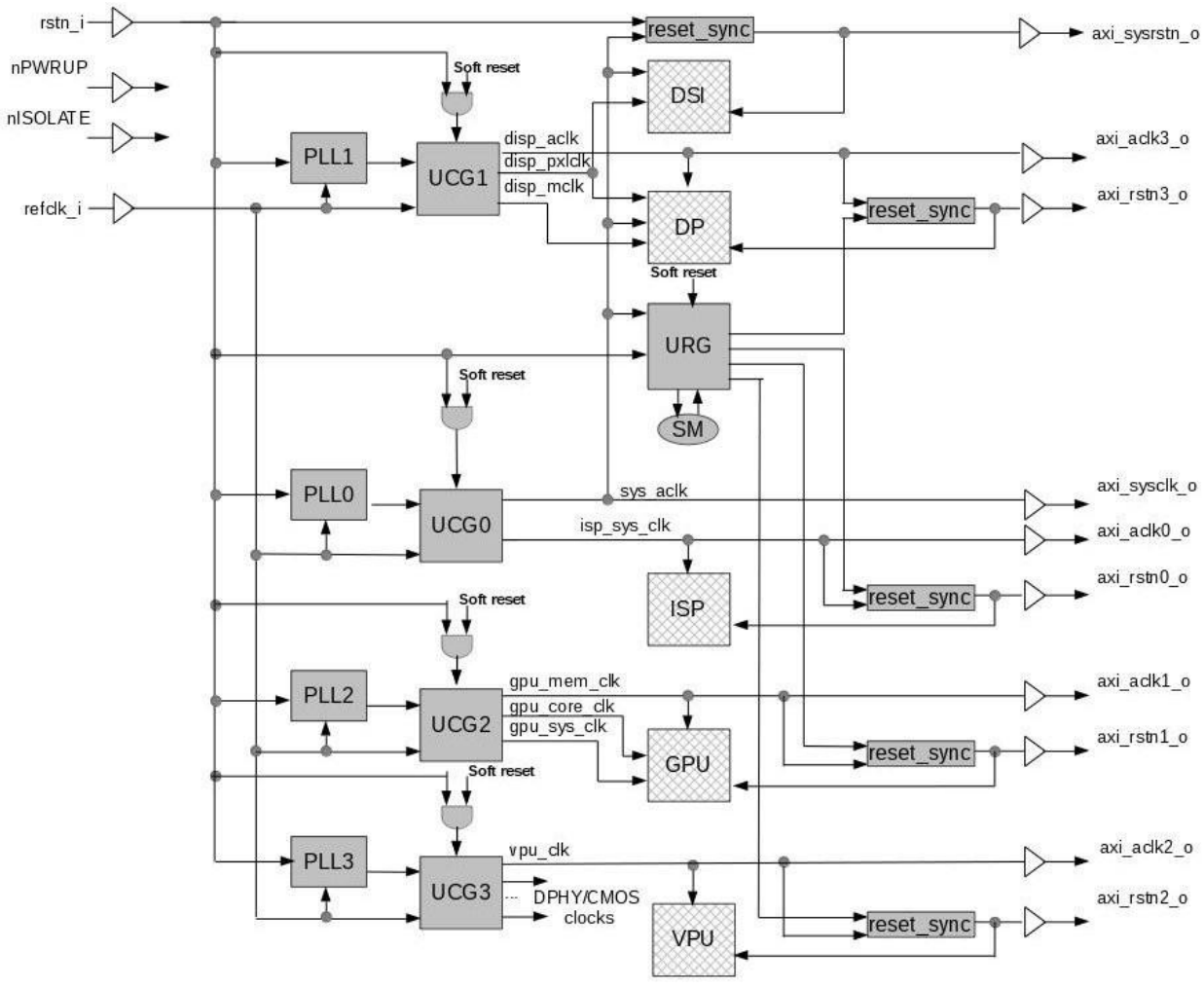


Рисунок 7.2. Управление мультимедийной подсистемой

После включения подсистемы входная референсная частота поступает на выход мультиплексора, формирующего системный синхросигнал sys_aclk (см. **Рисунок** 7.2). Все блоки подсистемы становятся доступны для программирования по системной шине. Программная инициализация подсистемы включает следующие шаги:

1. Переключение системного синхросигнала на референсную частоту установкой бита XTI_EN регистра SYS_CLK_CTRL.

2. Настройка PLL0-3 на требуемую частоту программированием полей SEL регистров PLL_CTRL0-3.

3. Настройка конфигурационных параметров подсистемы (регистр SUBSYSTEM_CFG).

4. Установка коэффициентов деления референсной частоты в полях DIV_COEFF регистров *CLK_CTRL.

5. Переключение системного синхросигнала на выход PLL0 сбросом бита XTI_EN регистра SYS_CLK_CTRL.

6. Настройка диапазонов частот конфигурационного и high-speed синхросигналов MIPI CSI в регистрах MIPI_RXn_CLK_PARAM, где n=0-1(если предполагается использование интерфейсов MIPI CSI).

7. Настройка диапазонов частот конфигурационного и high-speed синхросигналов MIPI DSI в регистре MIPI_TX_CLK_PARAM (если предполагается использование интерфейсов MIPI DSI).

8. Установка ручного режима управления для всех синхросигналов кроме isp_sys_clk, vpu_clk, disp_aclk (бит LPI_EN всех регистров *CLK_CTRL кроме ISP_CLK_CTRL, VPU_CLK_CTRL, DISPLAY_ACLK_CTRL должен быть установлен в 0). Для системного синхросигнала sys_aclk ручной режим управления установлен по умолчанию после сброса.

9. Включение требуемых синхросигналов установкой в состояние логической единицы битов CLK_EN регистров *CLK_CTRL. Для синхросигналов с автоматическим режимом LowPower интерфейса (LPI_EN=1) перед включением синхросигнала бит QACTIVE_CTL_EN должен быть установлен в состояние логического нуля (реакция на QACTIVE запрещена).

10. Снятие сброса и shutdown MIPI CSI RX0/1 записью единиц в разряды MIPI_RXn_RSTZ, MIPI_RXn_SHUTDOWNZ регистрах MIPI_RX_RST_CTRL/ MIPI_TX_CLK_CTRL, где n=0-1 (если предполагается использование интерфейсов MIPI CSI RX).

11. Снятие сброса CMOS сенсоров установкой в состояние логической единицы битов CMOSn_RST регистра CMOS_RST_CTRL, где n=0-1 (при условии, что сигналы сброса, формируемые микросхемой, используются для сброса CMOS сенсоров).

Дополнительно возможно осуществить программный сброс блоков ISP, GPU, VPU DP записью кода PP_WARM_RESET в поле POLICY регистров ISP_PPOLICY, GPU_PPOLICY, VPU_PPOLICY, DP_PPOLICY соответственно. Для выхода из состояния программного сброса требуется записать код в поле POLICY регистров ISP_PPOLICY, GPU_PPOLICY, VPU_PPOLICY, DP_PPOLICY соответственно.

7.4.2 Схема синхронизации

На вход подсистемы поступает синхросигнал refclk_i частотой 27.456 МГц. Для формирования сетки частот подсистемы используются четыре блока PLL (PLL0, PLL1, PLL2, PLL3). В качестве референсной частоты для всех блоков PLL используется синхросигнал refclk_i. Управление частотами на выходе PLL0,1,2 и 3 осуществляется программированием полей SEL регистров PLL_CTRL0,1,2 и 3 соответственно. Поле SEL задает коэффициент умножения референсной частоты от 1 до 128. При SEL=0 включается режим bypass PLL.

Синхросигналы с выходов PLL поступают на универсальные генераторы синхросигналов UCG0-3 (см. Рисунок 7.3).

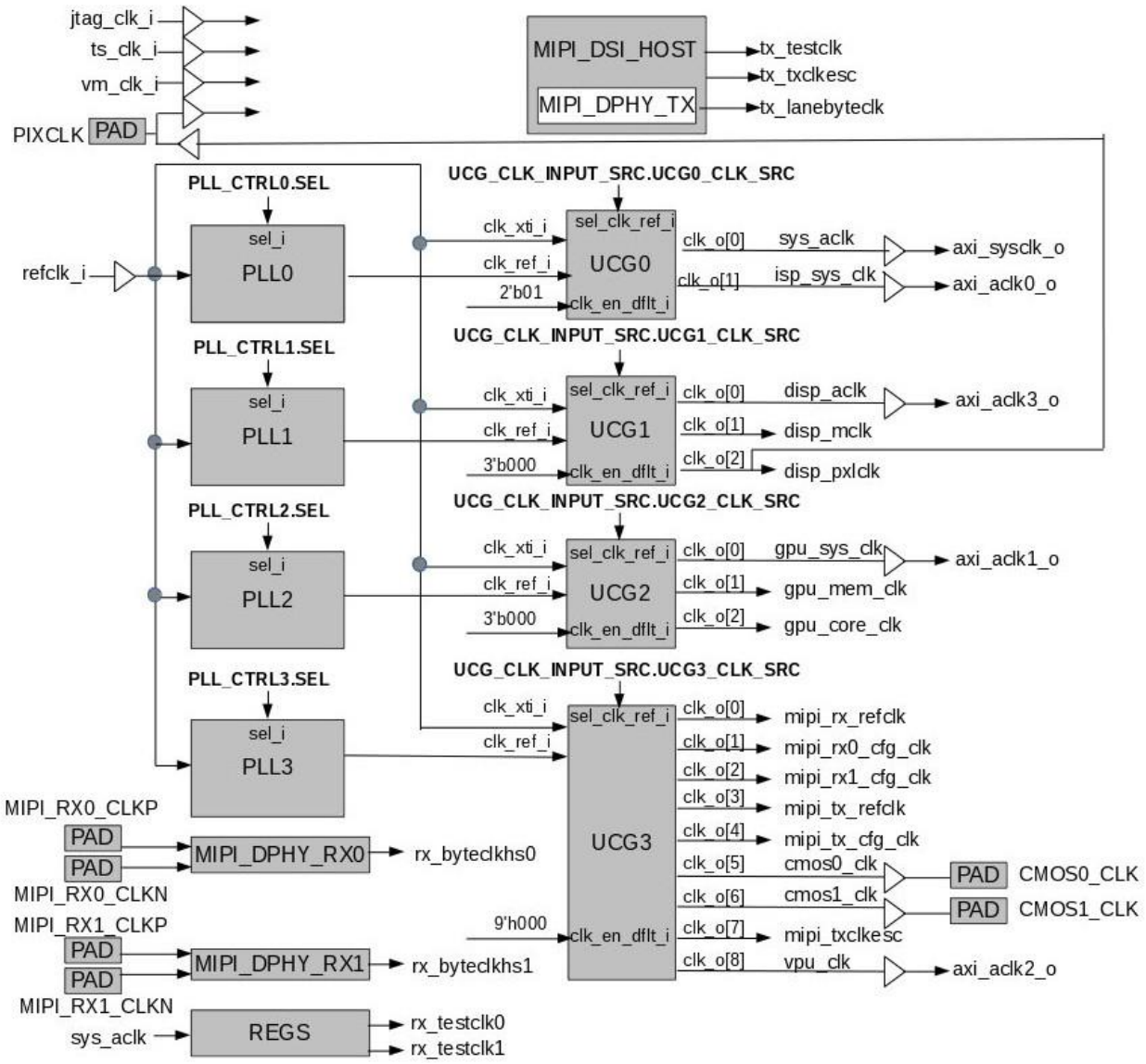


Рисунок 7.3. Схема синхронизации подсистемы

Управление выходными синхросигналами UCG0 осуществляется программированием регистров SYS_CLK_CTRL и ISP_CLK_CTRL.

Блок UCG0 формирует системный синхросигнал axi_sysclk_o. После включения питания на выход axi_sysclk_o выдается синхросигнал с выхода блока PLL0, который находится в режиме BYPASS. Источник системного синхросигнала может быть переключен на референсный синхросигнал refclk_i установкой бита XTI_EN регистра SYS_CLK_CTRL. Включение/выключение системного синхросигнала должно осуществляться в ручном режиме установкой/сбросом бита CLK_EN регистра SYS_CLK_CTRL. После сброса CLK_EN = 1. Бит LPI_EN регистра SYS_CLK_CTRL всегда должен быть установлен в состояние логического нуля.

Блок UCG0 формирует также синхросигнал блока isp (isp_sys_clk). Этот синхросигнал выводится на выход axi_aclk0 для использования их в качестве внешнего синхросигнала порта AXI Master 0. Включение/выключение синхросигнала должно осуществляться в

автоматическом режиме установкой/сбросом бита CLK_EN регистра ISP_CLK_CTRL при установленном бите LPI_EN и сброшенном бите QACTIVE_CTL_EN .

Блок UCG1 формирует синхросигналы контроллера дисплея (disp_aclk, disp_mclk, disp_pxlclk). Синхросигнал disp_aclk выводится на выход axi_aclk3_o для использования в качестве внешнего синхросигнала порта AXI Master 3. Синхросигнал disp_pxlclk выводится на контактную площадку PXLCLK использования в качестве внешнего синхросигнала параллельного порта вывода данных. Управление выходными синхросигналами UCG1 осуществляется программированием регистров DISPLAY_ACLK_CTRL, DISPLAY_MCLK_CTRL, DISPLAY_PXLCLK_CTRL. Включение/выключение синхросигналов должно осуществляться в автоматическом режиме установкой/сбросом бита CLK_EN регистров DISPLAY_ACLK_CTRL, DISPLAY_MCLK_CTRL и DISPLAY_PXLCLK_CTRL при установленном бите LPI_EN и сброшенном бите QACTIVE_CTL_EN .

Блок UCG2 формирует синхросигналы блока gpu (gpu_sys_clk, gpu_mem_clk, gpu_core_clk). Синхросигнал gpu_sys_clk выводится на выход axi_aclk1_o для использования в качестве внешнего синхросигнала порта AXI Master 1. Управление выходными синхросигналами UCG2 осуществляется программированием регистров GPU_SYS_CLK_CTRL, GPU_MEM_CLK_CTRL GPU_CORE_CLK_CTRL. Включение/выключение синхросигналов блока gpu должно осуществляться в ручном режиме установкой/сбросом бита CLK_EN регистров GPU_SYS_CLK_CTRL, GPU_MEM_CLK_CTRL, GPU_CORE_CLK_CTRL, а также программированием регистров блока gpu. Бит LPI_EN регистра SYS_CLK_CTRL всегда должен быть установлен в состояние логического нуля.

Последовательность включения синхросигналов блока gpu следующая:

1. Включить синхросигнал на выходе блока UCG2 установкой бита CLK_EN.
2. Включить синхросигнал внутри блока gpu с помощью регистров блока gpu.

Последовательность выключения синхросигналов блока gpu следующая:

1. Выключить синхросигнал внутри блока gpu с помощью регистров блока gpu.
2. Выключить синхросигнал на выходе блока UCG2 сбросом бита CLK_EN.

Блок UCG3 формирует синхросигналы блоков MIPI RX DPHY (mipi_rx_refclk, mipi_rx0_cfg_clk, mipi_rx1_cfg_clk и синхросигнал для производственного тестирования mipi_rx_txclkesc), и блока MIPI TX DPHY, размещенного внутри блока dsi_host_controller (mipi_tx_refclk и mipi_tx_cfg_clk). Блок UCG3 формирует также внешние синхросигналы CMOS сенсоров CMOS0_clk_o и CMOS1_clk_o. Управление выходными синхросигналами UCG2 осуществляется программированием регистров MIPI_RX_REFCLK_CTRL, MIPI_RX_CFG0_CLK_CTRL, MIPI_RX_CFG1_CLK_CTRL, MIPI_TXCLKESC_CLK_CTRL, MIPI_TX_REFCLK_CTRL, MIPI_TX_CFG_CLK_CTRL,

CMOS0_CLK_CTRL, CMOS1_CLK_CTRL. Включение/выключение синхросигналов MIPI DPHY и CMOS сенсоров должно осуществляться в ручном режиме установкой/сбросом бита CLK_EN вышеперечисленных регистров. Бит LPI_EN этих регистров всегда должен быть установлен в состояние логического нуля.

Блок UCG3 формирует также синхросигнал блока vpu (vpu_clk). Этот синхросигнал выводится на выход axi_aclk2 для использования их в качестве внешнего синхросигнала порта AXI Master 2. Включение/выключение синхросигнала должно осуществляться в автоматическом режиме установкой/сбросом бита CLK_EN регистра VPU_CLK_CTRL при установленном бите LPI_EN и сброшенном бите QACTIVE_CTL_EN.

Блоки приема видеоданных с портов MIPI CSI0/1 DPHY RX принимают высокоскоростной синхросигнал с clock lane. Частота этого синхросигнала находится в диапазоне от 1000 до 2500 МГц. Для передачи данных в ISP блоки MIPI CSI0/1 DHPY RX формируют синхросигналы с частотой в 8 раз ниже частоты входного высокочастотного синхросигнала (rx_byteclkhs0, rx_byteclkhs1).

Перечень синхросигналов подсистемы приведен в Таблица 7.39.

Таблица 7.39. Перечень синхросигналов подсистемы

Синхросигнал	Назначение	Частота (MHz)	Источник
Внешние источники синхронизации			
PIXCLK	Входной сихросигнал параллельного порта ввода видеоданных	150	Контактная площадка PIXCLK
refclk_i	Входная опорная частота	27.456	refclk_i
ts_clk_i	Синхросигнал последовательного интерфейса с температурным сенсором		ts_clk_i
vm_clk_i	Синхросигнал последовательного интерфейса с сенсором мониторинга напряжения		vm_clk_i
jtag_clk_i	Синхросигнал EJTAG порта блока gpu	40	jtag_clk_i
disp_axi_io_clk_phFdTx_i	Синхросигналы внешнего интерфейса контроллера дисплея	400	disp_axi_io_clk_phFdTx_i
disp_axi_io_clk_phBdRx_i		400	disp_axi_io_clk_phBdRx_i
gpu_axi_io_clk_phFdTx_i	Синхросигналы внешнего интерфейса gpu	400	gpu_axi_io_clk_phFdTx_i
gpu_axi_io_clk_phBdRx_i		400	gpu_axi_io_clk_phBdRx_i
isp_axi_io_clk_phFdTx_i	Синхросигналы внешнего интерфейса isp	400	isp_axi_io_clk_phFdTx_i
isp_axi_io_clk_phBdRx_i		400	isp_axi_io_clk_phBdRx_i
sys_io_clk_phFdRx_i	Синхросигналы внешнего системного интерфейса	400	sys_io_clk_phFdRx_i
sys_io_clk_phBdTx_i		400	sys_io_clk_phBdTx_i
vpu_axi_io_clk_phFdTx_i	Синхросигналы внешнего интерфейса vpu	400	vpu_axi_io_clk_phFdTx_i
vpu_axi_io_clk_phBdRx_i		400	vpu_axi_io_clk_phBdRx_i
Внутренние источники синхронизации			
sys_aclk	Системный синхросигнал	50 - 250	UCG0 ch0
isp_sys_clk	Синхросигнал блока isp	200 - 400	UCG0 ch1
vpu_clk	Синхросигнал блока vpu	75 - 600	UCG3 ch8

Синхросигнал	Назначение	Частота (MHz)	Источник
gpu_sys_clk	Системный синхросигнал блока gpu	19 - 600	UCG2 ch0
gpu_mem_clk	Синхросигнал подсистемы памяти блока gpu	19 - 600	UCG2 ch1
gpu_core_clk	Синхросигнал ядра блока gpu	19 - 600	UCG2 ch2
disp_aclk	Синхросигнал axi шины контроллера дисплея	50 - 600	UCG1 ch0
disp_mclk	Основной синхросигнал контроллера дисплея	6 – 312.5	UCG1 ch1
disp_pxlclk	Пиксельный синхросигнал контроллера дисплея	6 – 312.5	UCG1 ch2
mipi_rx_refclk	Опорная частота для MIPI RX DPHY PLL	2 - 64	UCG3 ch0
mipi_rx0_cfg_clk	Конфигурационный синхросигнал для MIPI RX0 DPHY	17 - 52	UCG3 ch1
mipi_rx1_cfg_clk	Конфигурационный синхросигнал для MIPI RX1 DPHY	17 - 52	UCG3 ch2
mipi_tx_refclk	Опорная частота для MIPI TX DPHY PLL	2 - 64	UCG3 ch3
mipi_tx_cfg_clk	Конфигурационный синхросигнал для MIPI TX DPHY	17 - 52	UCG3 ch4
mipi_txclkesc	Синхросигнал для производственных тестов в режиме low power	2 - 20	UCG3 ch7
cmos0_clk	Синхросигнал для CMOS сенсора 0	20 - 30	UCG3 ch5
cmos1_clk	Синхросигнал для CMOS сенсора 1	20 -30	UCG3 ch6
rx_byteclkhs0	Синхросигнал от MIPI CSI 0 DPHY приемника	125 - 312.5 (1000/8 - 2500/8)	u_mipi4_rx_dphy0/mipi_core_inst/rxbyteclkhs_if
rx_byteclkhs1	Синхросигнал от MIPI CSI 1 DPHY приемника	125 - 312.5 (1000/8 - 2500/8)	u_mipi4_rx_dphy1/mipi_core_inst/rxbyteclkhs_if
rx_testclk0	Синхросигнал для регистров DPHY приемника 0*	10	u_regs/MIPI_RX0_TSTREG_CTRL_TESTCLK
rx_testclk1	Синхросигнал для регистров DPHY приемника 1*	10	u_regs/MIPI_RX0_TSTREG_CTRL_TESTCLK
Внутренние синхросигналы контроллера MIPI DSI с интегрированным MIPI DPHY TX			
tx_lanebyteclk	Синхросигнал от MIPI DSI DPHY передатчика	125 - 312.5	u_mipi_dsi_host/DWC_mipi_dsi_host/u_phy/i01dphy/mipi_core_inst/txbyteclksrc_if
tx_txclkesc	Синхросигнал escape mode dphy передатчика**	20	u_mipi_dsi_host/DWC_mipi_dsi_host/u_cm/u_cmu/u_txesc/divclk
tx_testclk	Синхросигнал для регистров dphy передатчика*	10	u_mipi_dsi_host/DWC_mipi_dsi_host/u_regbank/phy_tst_ctrl0[1]
Синхсигналы программирования UCG			
i_ucg0/clk_apb_i	Синхросигнал программирования UCG0	50 - 250	UCG0 ch0
i_ucg1/clk_apb_i	Синхросигнал программирования UCG1	50 - 250	UCG0 ch0
i_ucg2/clk_apb_i	Синхросигнал программирования UCG2	50 - 250	UCG0 ch0
i_ucg3/clk_apb_i	Синхросигнал программирования UCG3	50 - 250	UCG0 ch0

* - формируется программно записью в регистр

** - формируется делением tx_lanebyteclk

Пример настройки частот сигналов синхронизации для частоты refclk 27.456 MHz приведен в Таблица 7.40. Частота пиксельной синхронизации контроллеров дисплея и DSI disp_pxlclk рассчитывалась для видеопотока 3840 x 2160 (UltraHD 4K) 30 fps (полный размер изображения с учетом промежутков между строками и кадрами рассчитывался как 4566 x 2250).

Таблица 7.40. Пример настройки частот сигналов синхронизации

Блок PLL/refclk_i	Коэффициент умножения	Выходная частота PLL (MHz)	Блок UCG	Коэффициент деления	Выходная частота синхросигнала (MHz)	Наименование синхросигнала
PLL0	45	1235,52	UCG0	DIV = 3	411.84	isp_sys_clk
				DIV = 5	247.104	sys_aclk
PLL1	45	1235,52	UCG1	DIV = 4	308.88	disp_core
				DIV = 4	308.88	disp_pxlclk
				DIV = 3	411.84	disp_aclk
PLL2	20	549,12	UCG2	DIV = 1	549.12	gpu_mem_clk
				DIV = 1	549.12	gpu_core_clk
				DIV = 2	274.56	gpu_sys_clk
PLL3	22(21)	604.032 (576.576)	UCG3	BYPASS	27.456*	mipi_rx_refclk
				BYPASS	27.456**	mipi_rx0_cfg_clk
				BYPASS	27.456**	mipi_rx1_cfg_clk
				BYPASS	27.456*	mipi_tx_refclk
				BYPASS	27.456**	mipi_tx_cfg_clk
				DIV = 31	19.485	mipi_txclkesc
				BYPASS	27.456	cmos0_clk
				BYPASS	27.456	cmos1_clk
				DIV = 1	604.032 (576.576)	vpu_clk

* - для получения максимальной частоты (60.403 MHz) DIV=10

** - для получения максимальной частоты (50.336 MHz) DIV=12

Таблица 7.41. Пример настройки частот контроллера для дисплея для вывода стандартных разрешений через параллельный порт

Формат	V active	Int/Prog	Частота синхросигнала по CEA-861 (MHz)	Коэффициент умножения	Коэффициент деления	Частота PXCLK (MHz)	Отклонение от стандартной частоты (%)
Выходные форматы с частотой ~59.94 Hz							
1	480	Prog	25.175	35	38	25.24	0.26
2,3	480	Prog	27.000	100	101	27.129	0.48
4	720	Prog	74.175	38	7	74.37	0.26
5	1080	Int	74.175	-	-	-	-

Формат	V active	Int/Prog	Частота синхросигнала по CEA-861 (MHz)	Коэффициент умножения	Коэффициент деления	Частота PXCLK (MHz)	Отклонение от стандартной частоты (%)
6,7	480	Int	27.000	-	-	-	-
8,9	240	Prog	27.000	100	101	27.129	0.48
10,11	480	Int	54.000	-	-	-	-
12,13	240	Prog	54.000	99	50	54.252	0.47
14,15	480	Prog	54.000	99	50	54.252	0.47
16	1080	Prog	148.350	38	7	148.74	0.26
Выходные форматы с частотой ~60 Hz							
1	480	Prog	25.200	35	38	25.24	0.24
2,3	480	Prog	27.027	100	101	27.129	0.48
4	720	Prog	74.250	38	14	74.37	0.16
5	1080	Int	74.250	-	-	-	-
6,7	480	Int	27.027	-	-	-	-
8,9	240	Prog	27.027	100	101	27.129	0.48
10,11	480	Int	54.054	-	-	-	-
12,13	240	Prog	54.054	99	50	54.252	0.47
14,15	480	Prog	54.054	99	50	54.252	0.47
16	1080	Prog	148.500	38	7	148.74	0.16
Выходные форматы с частотой ~50 Hz							
17,18	576	Prog	27.000	100	101	27.129	0.48
19	720	Prog	74.250	38	14	74.37	0.16
20	1080	Int	74.250	-	-	-	-
21,22	576	Int	27.000	-	-	-	-
23,24	288	Prog	27.000	100	101	27.129	0.48
25,26	576	Int	54.000	-	-	-	-
27.28	288	Prog	54.000	99	50	54.252	0.47
29,30	288	Prog	54.000	99	50	54.252	0.47
31	1080	Prog	148.500	38	7	148.74	0.16
Выходные форматы с частотами ~23.97, 24, 25, 29.97, 30 Hz							
32	1080	Prog	74.175	38	7	74.37	0.26
32	1080	Prog	74.250	38	14	74.37	0.16
33	1080	Prog	74.250	38	14	74.37	0.16
34	1080	Prog	74.175	38	7	74.37	0.26
34	1080	Prog	74.250	38	14	74.37	0.16

7.4.3 Сброс подсистемы

На вход подсистемы поступает глобальный асинхронный сигнал сброса resetn_i (см. Рисунок 7.4). Системный сигнал сброса sys_resetn устанавливается в активное состояние

асинхронно, в неактивное состояние — синхронно с системным синхросигналом sys_aclk. Для использования в качестве внешнего системного сигнала сброса сигнал sys_resetn выводится на выход axi_sysrstn_o.

Формирование сигналов сброса для блоков ISP, GPU, VPU и DP осуществляется блоком URG. Сигналы локального сброса от URG вырабатываются синхронно с системным синхросигналом sys_aclk. При установке в активное состояние входа асинхронного сброса resetn_i все сигналы локального сброса local_rstn_i устанавливаются в активное состояние. После установки resetn_i в неактивное состояние выходы local_rstn_o[7:0] устанавливаются в неактивное состояние через 4 такта sys_aclk. Выходы local_rstn_o[0], local_rstn_o[2], local_rstn_o[4], local_rstn_o[6] используются для формирования сигналов сброса блоков ISP, GPU, VPU и DP соответственно, выходы local_rstn_o[1], local_rstn_o[3], local_rstn_o[5], local_rstn_o[7] используются в автоматах управления.

Сброс приемников видеосигнала MIPI RX CSI0/1 DPHY осуществляется программно установкой в состояние логического нуля разрядов MIPI_RX0_RSTZ/ MIPI_RX1_RSTZ регистра RST_CTRL. Сигнал сброса mipi_rx0_rstz/mipi_rx1_rstz для блоков DPHY является асинхронным, поэтому пересинхронизация не требуется.

Для осуществления программного сброса блока ISP следует записать код PP_WARM_RST в поле POLICY регистра ISP_PPOLICY. Для выхода из состояния программного сброса требуется записать код PP_ON в поле POLICY регистра ISP_PPOLICY. Сигнал сброса, формируемый на выходе блока URG local_rstn_o[0], поступает на схему синхронизации reset_sync, которая вырабатывает сигнал асинхронного сброса блока ISP isp_sync_rstn, Этот сигнал используется в качестве внешнего сигнала сброса порта AXI Master 0 и выводится на выход axi_rstn0_o. Состояние блока ISP отображается в поле PSTATUS регистра ISP_PSTATUS. После включения питания подсистемы до снятия входного асинхронного сигнала сброса блок находится в нерабочем состоянии и PSTATUS = 0. После выхода из нерабочего состояния PSTATUS = PP_WARM_RESET при активном состоянии сигнала сброса блока ISP, PSTATUS = PP_ON при неактивном состоянии сигнала сброса блока ISP.

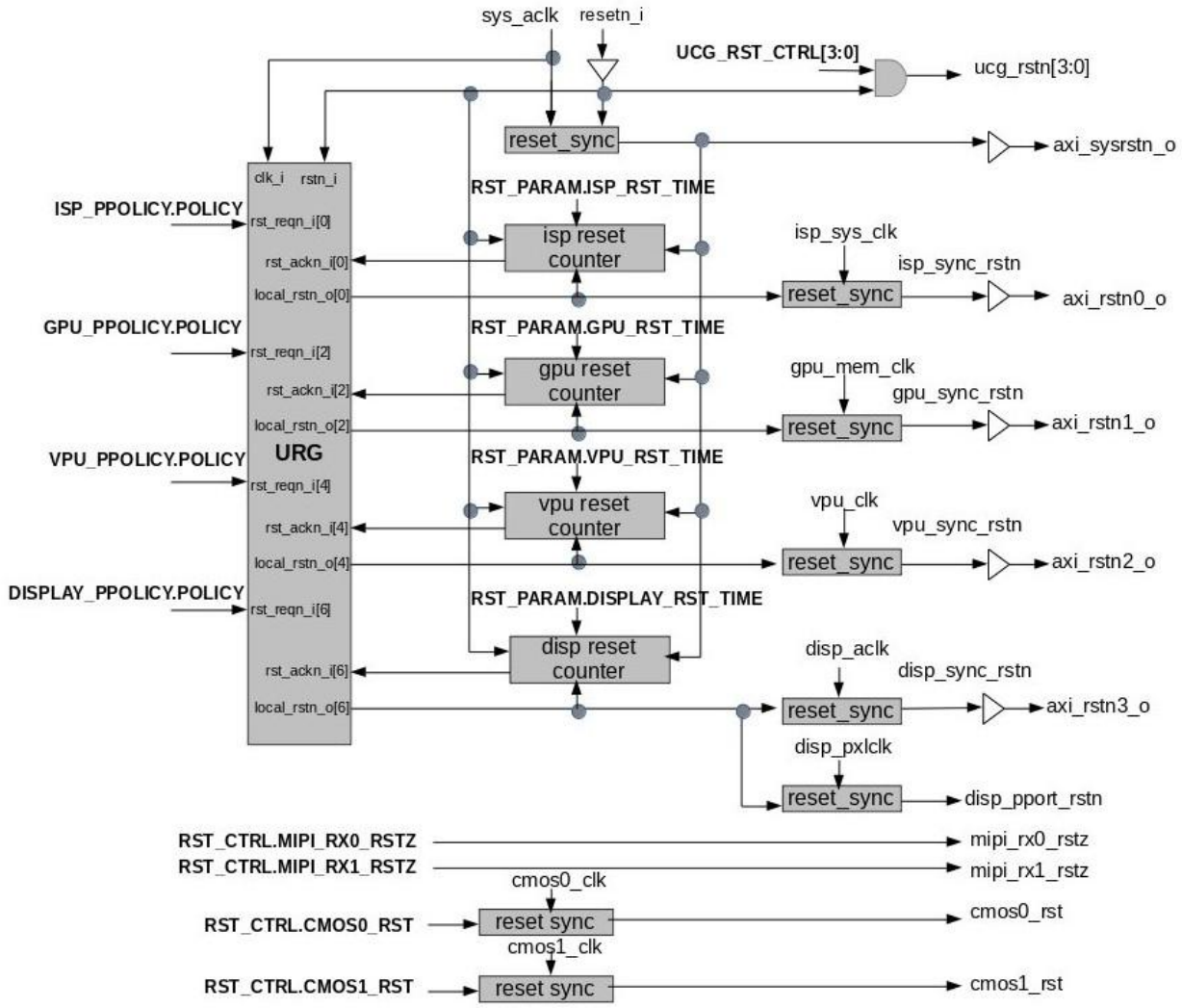


Рисунок 7.4. Схема сброса подсистемы

Для осуществления программного сброса блока GPU следует записать код PP_WARM_RST в поле POLICY регистра GPU_PPOLICY. Для выхода из состояния программного сброса требуется записать код PP_ON в поле POLICY регистра GPU_PPOLICY. Сигнал сброса, формируемый на выход блока URG local_rstn_o[2] поступает на схему синхронизации reset_sync, которая вырабатывает сигнал асинхронного сброса блока GPU gpu_sync_rstn. Этот сигнал используется в качестве внешнего сигнала сброса порта AXI Master 1 и выводится на выход axi_rstn1_o. Состояние блока GPU отображается в поле PSTATUS регистра GPU_PSTATUS. После включения питания подсистемы до снятия входного асинхронного сигнала сброса блок находится в нерабочем состоянии и PSTATUS = 0. После выхода из нерабочего состояния PSTATUS = PP_WARM_RESET при активном состоянии сигнала сброса блока GPU, PSTATUS = PP_ON при неактивном состоянии сигнала сброса блока GPU.

Для осуществления программного сброса блока VPU следует записать код PP_WARM_RST в поле POLICY регистра VPU_PPOLICY. Для выхода из состояния программного сброса требуется записать код PP_ON в поле POLICY регистра VPU_PPOLICY. Сигнал сброса,

формируемый на выход блока URG local_rstn_o[4] поступает на схему синхронизации reset_sync, которая вырабатывает сигнал асинхронного сброса блока VPU vpu_sync_rstn. Этот сигнал используется в качестве внешнего сигнала сброса порта AXI Master 2 и выводится на выход axi_rstn2_o. Состояние блока VPU отображается в поле PSTATUS регистра VPU_PSTATUS. После включения питания подсистемы до снятия входного асинхронного сигнала блок находится в нерабочем состоянии и PSTATUS = 0. После выхода из нерабочего состояния PSTATUS = PP_WARM_RESET при активном состоянии сигнала сброса блока VPU, PSTATUS = PP_ON при неактивном состоянии сигнала сброса блока VPU.

Для осуществления программного сброса блока DP следует записать код PP_WARM_RST в поле POLICY регистра DISPLAY_PPOLICY. Для выхода из состояния программного сброса требуется записать код PP_ON в поле POLICY регистра DISP_PPOLICY. Сигнал сброса, формируемый на выход блока URG local_rstn_o[6] поступает на схему синхронизации reset_sync, которая вырабатывает сигнал асинхронного сброса блока DP disp_sync_rstn. Для использования в качестве внешнего сигнала сброса порта AXI Master 3 сигнал disp_sync_rstn выводится на выход axi_rstn3_o. Для сброса параллельного порта дисплея формируется сигнал disp_pport_rstn, который функционально эквивалентен disp_sync_rstn, но устанавливается в неактивное состояние синхронно с синхросигналом disp_pxlclk. После включения питания подсистемы до снятия входного асинхронного сигнала блок находится в нерабочем состоянии и PSTATUS = 0. После выхода из нерабочего состояния PSTATUS = PP_WARM_RESET при активном состоянии сигнала сброса блока DP, PSTATUS = PP_ON при неактивном состоянии сигнала сброса блока DP.

Сигнал программного сброса универсальных генераторов синхросигналов UCG0-3 формируется при установке в состояние логической единицы разрядов UCG0_RST, UCG1_RST, UCG2_RST, UCG3_RST регистра UCG_RST_CTRL соответственно. Сигналы программного сброса блоков UCG объединяются по И с входным сигналом асинхронного сброса resetn_i.

Сброс приемников видеосигнала MIPI RX CSI0/1 DPHY осуществляется программно установкой в состояние логического нуля разрядов MIPI_RX0_RSTZ/ MIPI_RX1_RSTZ регистра MIPI_RX_RST_CTRL. Сигнал сброса mipi_rx0_rstz/mipi_rx1_rstz для блоков DPHY является асинхронным, поэтому пересинхронизация не требуется.

Сброс блоков управления CMOS сенсорами осуществляется при установке в состояние логической единицы разрядов CMOS0_RST/CMOS1_RST регистра RST_CTRL.

7.5 Скорости потоков данных

Таблица 7.42. Скорости потоков данных для UltraHD 4K

Блок	Запись по шине AXI			Чтение по шине AXI		
	Назначение	Тип потока	Гбит/сек	Тип потока	Размер потока	Гбит/сек
ISP	Выход Encoder	3840x2160 60fps YCbCr 422 8 bit (2 byte/pixel)	7.59			
	Выход Display	3840x2160 60fps RGB 8 bit (4 byte/pixel)	15.19			
Encoder	Bitstream	HEVC/H.264 (4 ядра 600 МГц)	0.39	Основной кадр	3840x2160 60fps YCbCr 422 8 bit (2 byte/pixel)	7.59
	Референсный кадр	3840x2160 60fps YCbCr 422 8 bit (2 byte/pixel)	7.59	Референсный кадр	3840x2160 60fps YCbCr 422 8 bit (2 byte/pixel)	7.59
Decoder	Декодированный кадр	3840x2160 60fps YCbCr 422 8 bit (2 byte/pixel)	7.59	Bitstream	HEVC/H.264 (6 ядер)	0.39
	Референсный кадр	3840x2160 60fps YCbCr 422 8 bit (2 byte/pixel)	7.59	Референсный кадр	3840x2160 60fps YCbCr 422 8 bit (2 byte/pixel)	7.59
Контроллер дисплея				Слой видео для вывода на экран	3840x2160 30fps RGB 8 bit (4 byte/pixel)	7.59
				Слой графика для вывода на экран	3840x2160 30fps RGB 8 bit (4 byte/pixel)	7.59
AXI0			22.78			0
AXI1			38.4			38.4
AXI2			23.16			23.16
AXI3			0			15.18

Примечание: Пропускная способность 128-разрядной AXI-шины 400 МГц = 50 Гбит/сек

8. ПОДСИСТЕМА DDR

8.1 Общая информация

Структурная схема подсистемы DDR представлена на Рисунок 8.1.

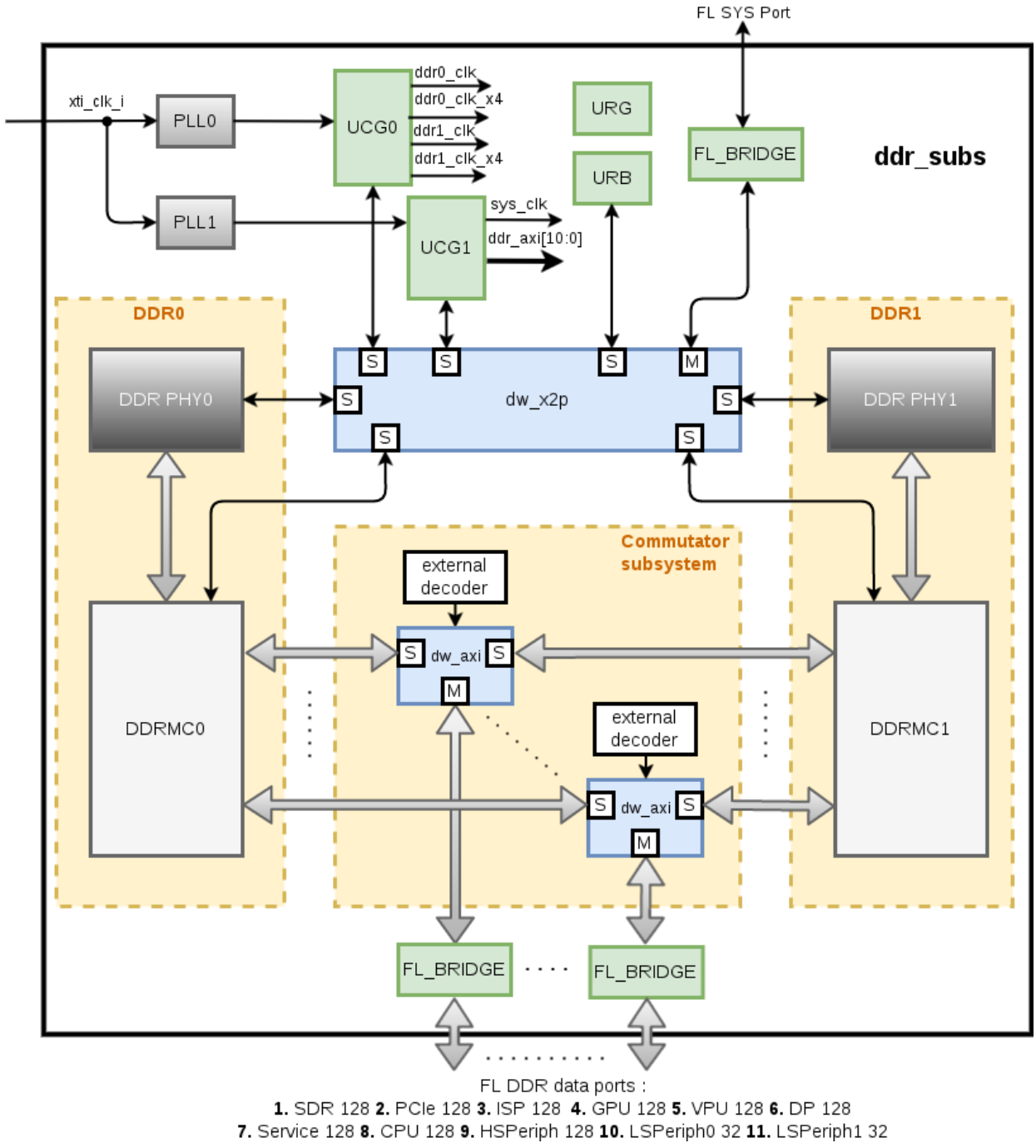


Рисунок 8.1. Структурная схема подсистемы DDR.

8.1.1 Состав подсистемы

Подсистема состоит из следующих модулей:

1. DDR PHY 0, DDR PHY1 – модули PHY DDR.
2. DDR0 uMCTL2, DDR1 uMCTL2 – контроллеры DDR. Контроллеры динамической памяти имеет следующий функционал:
    1) Поддержка протоколов DDR3, DDR3L, DDR4, LPDDR3, LPDDR4[1];
    2) Скорости передачи не менее 1600 Мбит/с;
    3) 32-разрядная шина данных;
    4) Адресация двух параллельно подключенных микросхем памяти с помощью сигнала chip select, в режиме DDR4 адресация до 4 блоков 3DS с помощью сигнала Chip ID, адресация до 64 Гбайт внешней памяти;
    5) Поддержка коррекции ошибок кодом Хэмминга.
3. PLL – системная PLL.
4. UCG – блок управления делителями частот подсистемы.
5. URG – блок управления сбросами элементов подсистемы.
6. URB – универсальных блок регистров. Содержит регистры управления подсистемой.
7. Commutator subsystem – совокупность коммутаторов, каждый из которых распределяет поток данных соответствующего AXI интерфейса подсистемы между двумя контроллерами DDR. Все коммутаторы поддерживают контроль пересечения транзакциями диапазона 4KB.
8. X2P – Коммутатор доступа к регистрам подсистемы.

8.2 Адресное пространство

Таблица 8.1. Адресное пространство

Смещение	Конечный адрес	Описание	Уровень доступа
0x4000000	0x5FFFFFF	DDR0 PHY Regs	Trusted, securable
0x6000000	0x600FFFF	DDR0 uMCTL2	
0x6010000	0x7FFFFFF	Резерв	
0x8000000	0x9FFFFFF	DDR1 PHY Regs	Trusted, securable
0xA000000	0xA00FFFF	DDR1 uMCTL2	
0xA010000	0xBFFFFFF	Резерв	
0xC000000	0xC00FFFF	URB	Trusted, securable
0xC010000	0xC01FFFF	UCG0	
0xC020000	0xC02FFFF	UCG1	

[1] Максимальный объем памяти LPDDR4, поддерживаемый К1892ВМ21Я, составляет 8 Гбайт на контроллер.

8.3 Схема синхронизации

Схема синхронизации подсистемы DDR представлена на Рисунок 8.2.

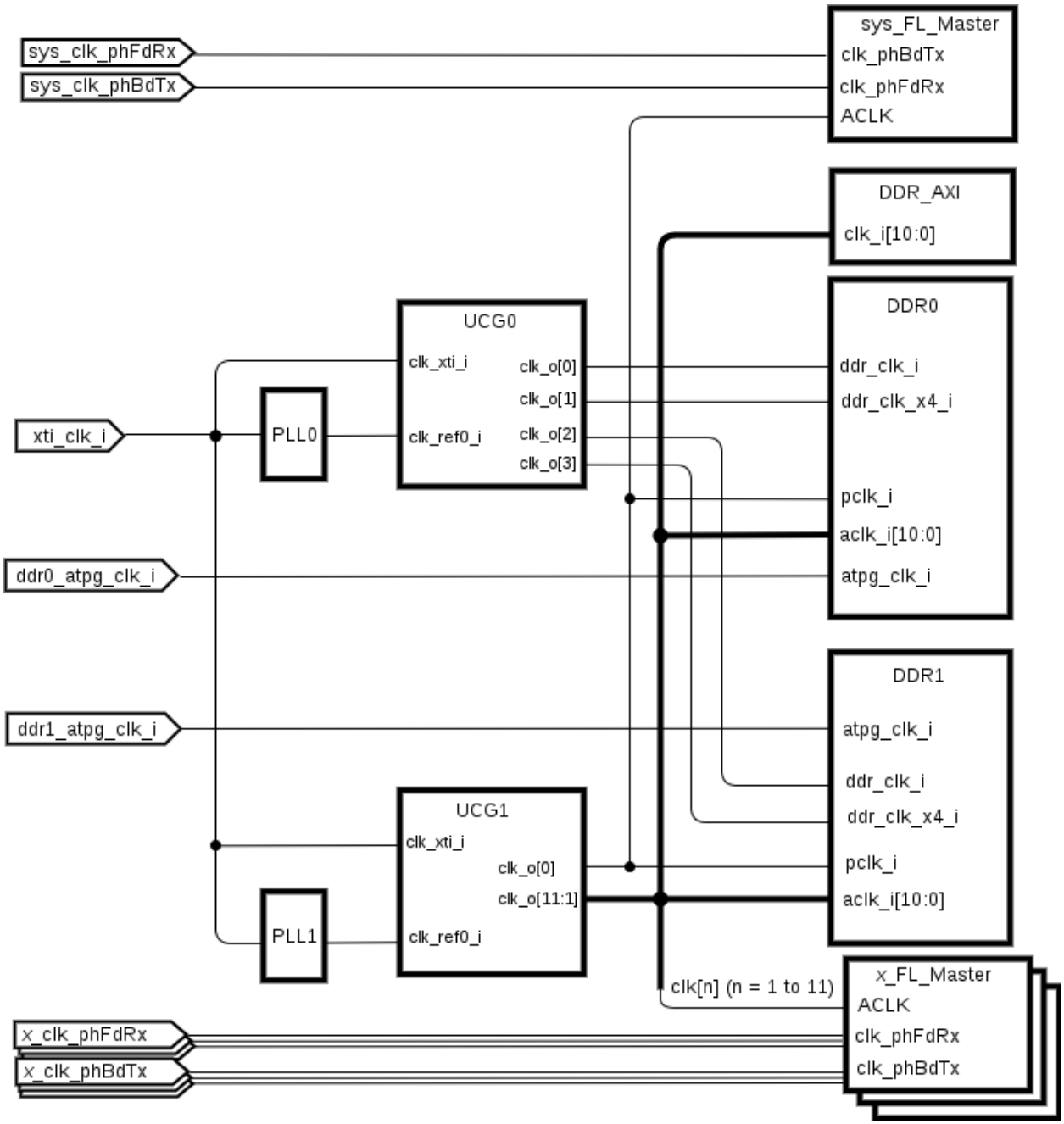


Рисунок 8.2. Схема синхронизации подсистемы DDR

Таблица 8.2. Источники тактовых сигналов ddr_subs. Частота xti_clk_i = 27.456 Mhz

Название	Источник	Частота (Mhz)	Коэффициент умножения PLL	Коэффициент деления UCG
xti_clk	xti_clk_i	27.456		
ddr0_atpg_clk_i	Генератор тестовых векторов (DFT)	До 3.2 ГГц		

Название	Источник	Частота (Mhz)	Коэффициент умножения PLL	Коэффициент деления UCG
ddr1_atpg_clk_i	Генератор тестовых векторов (DFT)	До 3.2 ГГц		
bs_clock_i	Генератор векторов Boundary Scan	100 МГц		
ddr0_clk	UCG0.clk_o[0]	800 (794.6)	29	1
ddr0_clk_x4	UCG0.clk_o[1]	667	24	1
ddr1_clk	UCG0.clk_o[2]	800 (794.6)	29	1
ddr1_clk_x4	UCG0.clk_o[3]	667 (657.6)	24	1
sys_clk	UCG1.clk_o[0]	200	44	
sdr_clk	UCG1.clk_o[1]	600	44	2
pcie_clk	UCG1.clk_o[2]	600	44	2
isp_clk	UCG1.clk_o[3]	200	44	6
gpu_clk	UCG1.clk_o[4]	600	44	2
vpu_clk	UCG1.clk_o[5]	300	44	4
dp_clk	UCG1.clk_o[6]	400	44	3
cpu_clk	UCG1.clk_o[7]	600	44	2
service_clk	UCG1.clk_o[8]	300	44	4
hsperiph_clk	UCG1.clk_o[9]	400	44	3
lsperiph0_clk	UCG1.clk_o[10]	200	44	6
lsperiph1_clk	UCG1.clk_o[11]	200	44	6
sys_clk_fhFdRx	sys_io_clk_fhFdRx_i	200	см. Раздел 3.1.1.	
sys_clk_fhBdT	sys_io_clk_fhBdTx_i	200		
sdr_clk_fhFdRx	sdr_dsp_io_clk_fhFdRx_i	600		
sdr_clk_fhBdTx	sdr_dsp_io_clk_fhBdTx_i	600		
pcie_clk_fhFdRx	sdr_pcie_io_clk_fhFdRx_i	600		
pcie_clk_fhBdTx	sdr_pcie_io_clk_fhBdTx_i	600		
isp_clk_fhFdRx	isp_io_clk_fhFdRx_i	200		
isp_clk_fhBdTx	isp_io_clk_fhBdTx_i	200		
gpu_clk_fhFdRx	gpu_io_clk_fhFdRx_i	600		
gpu_clk_fhBdTx	gpu_io_clk_fhBdTx_i	600		
vpu_clk_fhFdRx	vpu_io_clk_fhFdRx_i	300		
vpu_clk_fhBdTx	vpu_io_clk_fhBdTx_i	300		
dp_clk_fhFdRx	dp_io_clk_fhFdRx_i	400		
dp_clk_fhBdTx	dp_io_clk_fhBdTx_i	400		
cpu_clk_fhFdRx	cpu_io_clk_fhFdRx_i	600		
cpu_clk_fhBdTx	cpu_io_clk_fhBdTx_i	600		
service_clk_fhFdRx	service_io_clk_fhFdRx_i	300		
serivce_clk_fhBdTx	serivce_io_clk_fhBdTx_i	300		
hsperiph_clk_fhFdRx	hsperiph_io_clk_fhFdRx_i	400		
hsperiph_clk_fhBdTx	hsperiph_io_clk_fhBdTx_i	400		
lsperiph0_clk_fhFdRx	lsperiph0_io_clk_fhFdRx_i	200		
lsperiph0_clk_fhBdTx	lsperiph0_io_clk_fhBdTx_i	200		
lsperiph1_clk_fhFdRx	lsperiph1_io_clk_fhFdRx_i	200		
lsperiph1_clk_fhBdTx	lsperiph1_io_clk_fhBdTx_i	200		

Каждой выходной частоте UCG соответствует Low Power интерфейс Q-channel. Более подробно об этом можно прочитать в Главе 43 «Универсальный блок формирования тактовой частоты (UCG)».

8.4 Схема управления сбросами

Схема управления сбросами подсистемы DDR представлена на Рисунок 8.3.

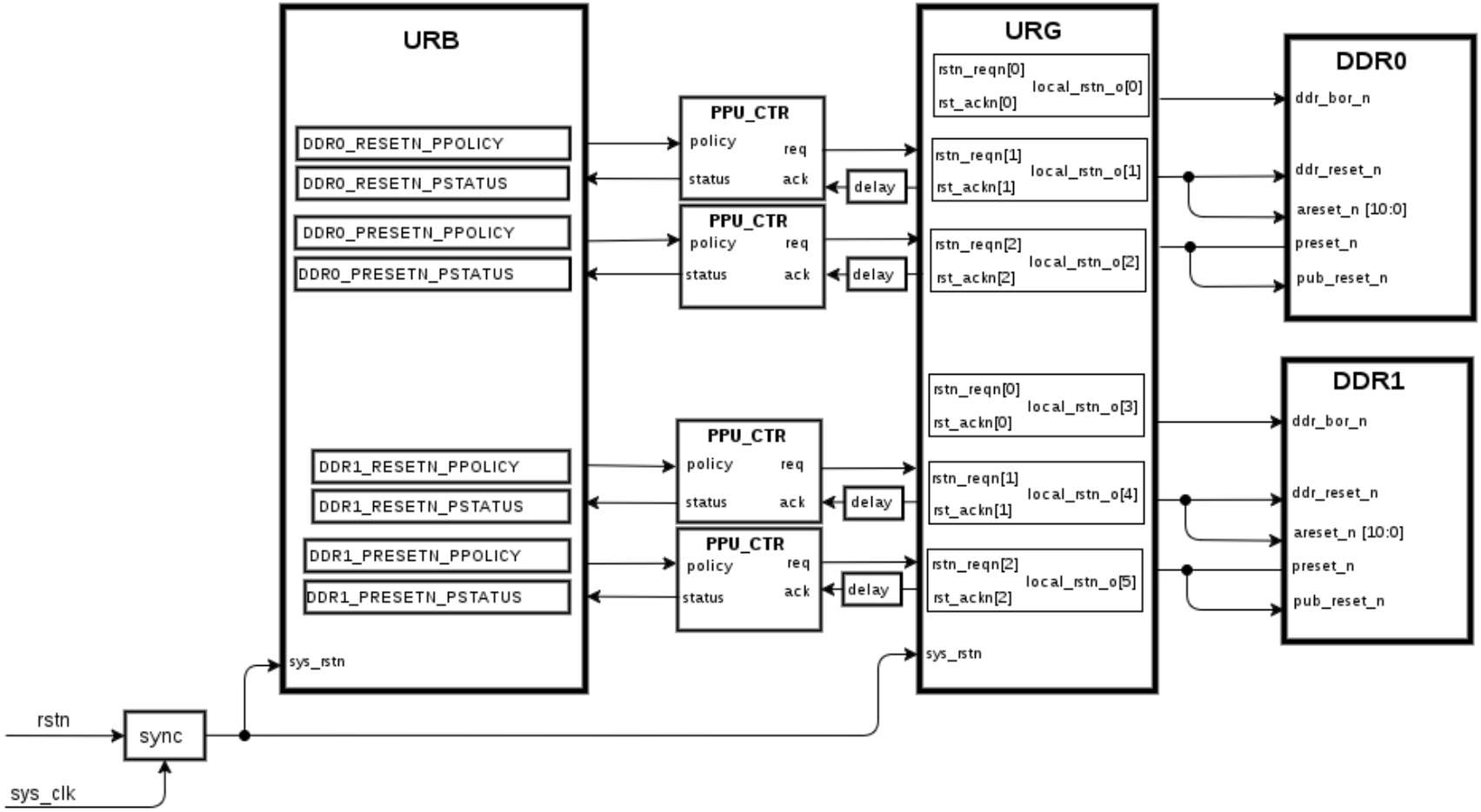


Рисунок 8.3. Схема управления сбросами подсистемы DDR

Операции записи в регистры управления сбросом допускаются только при условии работы UCG0 подсистемы в bypass mode (UCG_BP_CTR_REG=0xFFFF).

Описание выхода подсистемы DDR из системного сброса:

1. Все коммутационные блоки подсистемы (commutator subsystem, dw_x2p) выйдут из сброса вместе с rstn. Также будет сняты асинхронные сбросы ddr_bor_n обоих PHY (PUB).
2. После выхода необходимо программно снять сигналы сбросов PUB и APB интерфейсов сборок: pub_resetn_n и present. Если значение регистра ddr*_presetn_pstatus соответствует состоянию “ON”, значит возможет тренинг внешней памяти и программирование контроллера памяти.
3. Выход из сбросов ddr_reset_n и aresetn производет при соблюдении 2 условий:
   1) Со времени выхода из системного сброса rstn прошло 64 такта частоты sys_clk.
   2) Соотсвтующий регистр ddr*_reset_ppolicy был переведен в состояние “ON”.

   После того, как регистр ddr*_reset_pstatus перейдет в состояние “ON” – сборка готова к работе.

Реализация теплого сброса:

1. Для того, чтобы отправить одну из сборок DDRMC+PHY(PUB) в теплый сброс, необходимо перевести в состояние WARM_RESET сначала регистр ddr*_resetn_ppolicy, а затем ddr*_presetn_ppolicy.
2. Убедившись, что регистры ddr*_resetn_pstatus и ddr*_presetn_pstatus находятся в состоянии WARM_RESET, следует перевести регистр ddr*_presetn_ppolicy в состояние “ON”. При этом, выход из pub_reset_n и present произойдет только при условии, что их продолжительность была не менее 16-и тактов.
3. Если значение регистра ddr*_presetn_pstatus соответствует состоянию “ON”, значит возможет тренинг внешней памяти и программирование контроллера памяти.
4. Выход из сбросов ddr_reset_n и aresetn производет при соблюдении 2 условий:
   1) Со времени выхода из системного сброса rstn прошло 64 такта частоты sys_clk.
   2) Соотсвтующий регистр ddr*_reset_ppolicy был переведен в состояние “ON”.

   После того, как регистр ddr*_reset_pstatus перейдет в состояние “ON” – сборка готова к работе.

8.5 Регистры

Перечень регистров управления подсистемой приведен в Таблица 8.3, Таблица 8.4, Таблица 8.5.

Таблица 8.3. Перечень программно-доступных регистров UCG0

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
UCG_CTR00_REG	Регистр управления каналом ddr0_clk	0x0	W/R	0x0
UCG_CTR01_REG	Регистр управления каналом ddr0_clk_x4	0x0	W/R	0x4
UCG_CTR02_REG	Регистр управления каналом ddr1_clk	0x0	W/R	0x8
UCG_CTR03_REG	Регистр управления каналом ddr1_clk_x4	0x0	W/R	0xC

Таблица 8.4. Перечень программно-доступных регистров UCG1

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
UCG_CTR00_REG	Регистр управления каналом sys_clk	0x0	W/R	0x0
UCG_CTR01_REG	Регистр управления каналом axi_clk[0] (sdr_clk)	0x0	W/R	0x4
UCG_CTR02_REG	Регистр управления каналом axi_clk[1] (pcie_clk)	0x0	W/R	0x8
UCG_CTR03_REG	Регистр управления каналом axi_clk[2] (isp_clk)	0x0	W/R	0xC
UCG_CTR04_REG	Регистр управления каналом axi_clk[3] (gpu_clk)	0x0	W/R	0x10
UCG_CTR05_REG	Регистр управления каналом axi_clk[4] (vpu_clk)	0x0	W/R	0x14
UCG_CTR06_REG	Регистр управления каналом axi_clk[5] (dp_clk)	0x0	W/R	0x18
UCG_CTR07_REG	Регистр управления каналом axi_clk[6] (cpu_clk)	0x0	W/R	0x1C
UCG_CTR08_REG	Регистр управления каналом axi_clk[7] (service_clk)	0x0	W/R	0x20
UCG_CTR09_REG	Регистр управления каналом axi_clk[8] (hsperiph_clk)	0x0	W/R	0x24
UCG_CTR10_REG	Регистр управления каналом axi_clk[9] (lsperiph0_clk)	0x0	W/R	0x28
UCG_CTR11_REG	Регистр управления каналом axi_clk[10] (lsperiph1_clk)	0x0	W/R	0x2C

Таблица 8.5. Перечень программно-доступных регистров URB

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
PLLCFG0	Регистр конфигурации PLL0	0x0	W/R	0x0
PLLDIAG0	Регистр диагностики PLL0	0x0	W/R	0x4
PLLCFG1	Регистр конфигурации PLL1	0x0	W/R	0x8
PLLDIAG1	Регистр диагностики PLL1	0x0	W/R	0xC
XDECODER_MODE	Регистр выбора режима работы декодеров AXI-коммутаторов	0x0	W/R	0x10
DDR_AXI_CHS_ENABLE	Регистр включения AXI интерфейсов доступа к памяти.	0x0	W/R	0x14
DDR0_HIF_MRR_DATA_0	DDR0 hif_mrr_data [31:0]	0x0	R	0x20
DDR0_HIF_MRR_DATA_1	DDR0 hif_mrr_data [63:32]	0x0	R	0x24
DDR0_HIF_MRR_DATA_2	DDR0 hif_mrr_data [95:64]	0x0	R	0x28
DDR0_HIF_MRR_DATA_3	DDR0 hif_mrr_data [127:96]	0x0	R	0x2C

DDR1_HIF_MRR_DATA_0	DDR1 hif_mrr_data [31:0]	0x0	R	0x30
DDR1_HIF_MRR_DATA_1	DDR1 hif_mrr_data [63:32]	0x0	R	0x34
DDR1_HIF_MRR_DATA_2	DDR1 hif_mrr_data [95:64]	0x0	R	0x38
DDR1_HIF_MRR_DATA_3	DDR1 hif_mrr_data [127:96]	0x0	R	0x3C
DDR0_AWQOS_0	Нулевой регистр управления сигналами AWQOS AXI интерфейсов DDR0.	0x0	W/R	0x40
DDR0_AWQOS_1	Первый регистр управления сигналами AWQOS AXI интерфейсов DDR0.	0x0	W/R	0x44
DDR0_ARQOS_0	Нулевой регистр управления сигналами ARQOS AXI интерфейсов DDR0.	0x0	W/R	0x48
DDR0_ARQOS_1	Первый регистр управления сигналами ARQOS AXI интерфейсов DDR0.	0x0	W/R	0x4C
DDR1_AWQOS_0	Нулевой регистр управления сигналами AWQOS AXI интерфейсов DDR1.	0x0	W/R	0x50
DDR1_AWQOS_1	Первый регистр управления сигналами AWQOS AXI интерфейсов DDR1.	0x0	W/R	0x54
DDR1_ARQOS_0	Нулевой регистр управления сигналами ARQOS AXI интерфейсов DDR1.	0x0	W/R	0x58
DDR1_ARQOS_1	Первый регистр управления сигналами ARQOS AXI интерфейсов DDR1.	0x0	W/R	0x5C
DDR0_RESETN_PPOLICY	Регистр установки состояния контроллера DDR0.	0x8	W/R	0x60
DDR0_RESETN_PSTATUS	Регистр статуса состояния контроллера DDR0	0x8	R	0x64
DDR0_PRESETN_PPOLICY	Регистр установки состояния PHY DDR0.	0x0	W/R	0x68
DDR0_PRESETN_PSTATUS	Регистр статуса состояния я PHY DDR0	0x0	R	0x6C
DDR1_RESETN_PPOLICY	Регистр установки состояния контроллера DDR1.	0x8	W/R	0x70
DDR1_RESETN_PSTATUS	Регистр статуса состояния контроллера DDR1	0x8	R	0x74
DDR1_PRESETN_PPOLICY	Регистр установки состояния PHY DDR1.	0x0	W/R	0x78
DDR1_PRESETN_PSTATUS	Регистр статуса состояния PHY DDR1	0x0	R	0x7C
CTRSECUREREGION	Регистр управления доступом к выделенным адресным диапазонам памяти.	0x0	W/R	0x80
BASESECUREREGION0_ LOW	База адресного диапазона 0. Младшая часть.	0x0	W/R	0x90
BASESECUREREGION0_ HIGH	База адресного диапазона 0. Старшая часть	0x0	W/R	0x94
MASKSECUREREGION0_ LOW	Маска адресного диапазона 0. Младшая часть.	0x0	W/R	0x98
MASKSECUREREGION0_ HIGH	Маска адресного диапазона 0. Старшая часть.	0x0	W/R	0x9C
BASESECUREREGION1_ LOW	База адресного диапазона 1. Младшая часть.	0x0	W/R	0xA0
BASESECUREREGION1_ HIGH	База адресного диапазона 1. Старшая часть	0x0	W/R	0xA4

MASKSECUREREGION1_ LOW	Маска адресного диапазона 1. Младшая часть.	0x0	W/R	0xA8
MASKSECUREREGION1_ HIGH	Маска адресного диапазона 1. Старшая часть.	0x0	W/R	0xAC
BASESECUREREGION2_ LOW	База адресного диапазона 2. Младшая часть.	0x0	W/R	0xB0
BASESECUREREGION2_ HIGH	База адресного диапазона 2. Старшая часть	0x0	W/R	0xB4
MASKSECUREREGION2_ LOW	Маска адресного диапазона 2. Младшая часть.	0x0	W/R	0xB8
MASKSECUREREGION2_ HIGH	Маска адресного диапазона 2. Старшая часть.	0x0	W/R	0xBC
BASESECUREREGION3_ LOW	База адресного диапазона 3. Младшая часть.	0x0	W/R	0xC0
BASESECUREREGION3_ HIGH	База адресного диапазона 3. Старшая часть	0x0	W/R	0xC4
MASKSECUREREGION3_ LOW	Маска адресного диапазона 3. Младшая часть.	0x0	W/R	0xC8
MASKSECUREREGION3_ HIGH	Маска адресного диапазона 3. Старшая часть.	0x0	W/R	0xCC
DDR0_RAM_SLEEP	Регистр перевода памятей DDR0 в режим пониженного энергопотребления	0x0	W/R	0xD0
DDR1_RAM_SLEEP	Регистр перевода памятей DDR1 в режим пониженного энергопотребления	0x0	W/R	0xD4
GPU_BAR	Биты [35:32] интерфейса GPU AXI.	0x1E	W/R	0xD8
HSPERPIH_BAR	Биты [35:32] интерфейса HSPERPIH AXI.	0x1E	W/R	0xDC
LSPERIPH0_BAR	Биты [35:32] интерфейса LSPERPIH0 AXI.	0x1E	W/R	0xE0
LSPERIPH1_BAR	Биты [35:32] интерфейса LSPERIPH1 AXI.	0x1E	W/R	0xE4

О регистрах UCG более подробно можно прочитать в Главе 43.

8.5.1 Регистр PLLCFG*x*

Регистр конфигурации PLL*x*.

Формат регистра PLLCFGx приведен в следующей таблице:

Таблица 8.6. Поля регистра PLLCFG*x*

Диапазон	Название	Описание	Сброс
7:0	sel	Определяет выходную частоту и режим работы PLL: 0x0 - опорная частота ref_clk, PLL выключена; 0x1 - ref_clk * 2; 0x2 - ref_clk * 3; 0x3 - ref_clk * 4; ... - ...; 0x73 - ref_clk * 116; 0x74 - ref_clk * 116; ... - ...; 0xFF - ref_clk * 116; При sel > 0 и man = 1, конфигурация pll определяется полями *_man данного регистра	
8:8	-	Резерв	0

Диапазон	Название	Описание	Сброс
9:9	man	При бите заданном в 1 и sel > 0 конфигурация pll определяется полям *_man данного регистра. Выходная частота определяется по формуле ref_clk * (nf_man + 1) / (nr_man + 1) / (od_man + 1). Поля *_man рекомендуется задавать исходя из значений генерируемым скриптом настройки pll. Значение выходной частоты не должно превышать 3600 МГц.	0
13:10	od_man	Коэффициент od настройки pll	0
26:14	nf_man	Коэффициент nf настройки pll	0
30:27	nr_man	Коэффициент nr настройки pll	
31	lock	Бит указывает окончание настройки PLL на новую заданную частоту работы. При начале процедуры настройки либо при выключенной PLL сбрасывается в 0.	0

8.5.2 Регистр PLLDIAGx

Регистр диагностики PLL*x*.

Формат регистра PLLDIAG*x* приведен в следующей таблице:

Таблица 8.7. Поля регистра PLLDIAGx

Диапазон	Название	Описание	Сброс
0:0	test	Диагностический вход pll - bypass опорной частоты pll на выход при 1	0
1:1	ensat	Диагностический вход pll - разрешает насыщение, в нормальном режиме - 1	1
2:2	fasten	Диагностический вход pll - разрешает механизм fast lock	0
3:3	rfslip	Диагностический выход pll - регистрирует ситуацию referency cycle slip	0
4:4	fbslip	Диагностический выход pll - регистрирует ситуацию feedback cycle slip	0
31:5	-	Резерв	

8.5.3 Регистр XDECODER_MODE

Регистр выбора режима работы декодеров AXI-коммутаторов commutator subsystem.

Формат регистра XDECODER_MODE приведен в Таблица 8.8.

Таблица 8.8. Формат регистра XDECODER_MODE

Номер бита	Условное обозначение	Назначение
1	XDECODER_MODE	Регистр выбора режима адресации: 0: Режим прямой адресации 1: Режим чередования адреса Более подробно см. Раздел 3.1. Карта памяти микросхемы
31:1	-	Резерв

8.5.4 Регистр DDR_AXI_CHS_ENABLE

Регистр включения AXI-интерфейсов доступа к памяти.

Формат регистра DDR_AXI_CHS_ENABLE приведен в Таблица 8.9.

Таблица 8.9. Формат регистра DDR_AXI_CHS_ENABLE

Номер бита	Условное обозначение	Назначение
0	SDR_DDR0_ENABLE	Единица в этом бите разрешает интерфейсу sdr доступ к контроллеру ddr0.
1	PCIE_DDR0_ENABLE	Единица в этом бите разрешает интерфейсу pcie доступ к контроллеру ddr0.
2	ISP_DDR0_ENABLE	Единица в этом бите разрешает интерфейсу isp доступ к контроллеру ddr0.
3	GPU_DDR0_ENABLE	Единица в этом бите разрешает интерфейсу gpu доступ к контроллеру ddr0.
4	VPU_DDR0_ENABLE	Единица в этом бите разрешает интерфейсу vpu доступ к контроллеру ddr0.
5	DP_DDR0_ENABLE	Единица в этом бите разрешает интерфейсу dp доступ к контроллеру ddr0.
6	CPU_DDR0_ENABLE	Единица в этом бите разрешает интерфейсу cpu доступ к контроллеру ddr0.
7	SERVICE_DDR0_ ENABLE	Единица в этом бите разрешает интерфейсу service доступ к контроллеру ddr0.
8	HSPERIPH_DDR0_ ENABLE	Единица в этом бите разрешает интерфейсу hsperiph доступ к контроллеру ddr0.
9	LSPERIPH0_DDR0_ ENABLE	Единица в этом бите разрешает интерфейсу lsperiph0 доступ к контроллеру ddr0.
10	LSPERIPH1_DDR0_ ENABLE	Единица в этом бите разрешает интерфейсу lsperiph1 доступ к контроллеру ddr0.
15:11	-	Резерв
16	SDR_DDR1_ENABLE	Единица в этом бите разрешает интерфейсу sdr доступ к контроллеру ddr1.
17	PCIE_DDR1_ENABLE	Единица в этом бите разрешает интерфейсу pcie доступ к контроллеру ddr1.
18	ISP_DDR1_ENABLE	Единица в этом бите разрешает интерфейсу isp доступ к контроллеру ddr1.
19	GPU_DDR1_ENABLE	Единица в этом бите разрешает интерфейсу gpu доступ к контроллеру ddr1.
20	VPU_DDR1_ENABLE	Единица в этом бите разрешает интерфейсу vpu доступ к контроллеру ddr1.
21	DP_DDR1_ENABLE	Единица в этом бите разрешает интерфейсу dp доступ к контроллеру ddr1.
22	CPU_DDR1_ENABLE	Единица в этом бите разрешает интерфейсу cpu доступ к контроллеру ddr1.
23	SERVICE_DDR1_ ENABLE	Единица в этом бите разрешает интерфейсу service доступ к контроллеру ddr1.
24	HSPERIPH_DDR1_ ENABLE	Единица в этом бите разрешает интерфейсу hsperiph доступ к контроллеру ddr1.
25	LSPERIPH0_DDR1_ ENABLE	Единица в этом бите разрешает интерфейсу lsperiph0 доступ к контроллеру ddr1.
26	LSPERIPH1_DDR1_ ENABLE	Единица в этом бите разрешает интерфейсу lsperiph1 доступ к контроллеру ddr1.
31:22	-	Резерв

8.5.5 Регистр DDR0_RESETN_PPOLICY

Регистр установки состояния PHY DDR0.

Таблица 8.10. Формат регистра DDR0_RESETN_PPOLICY

Номер бита	Условное обозначение	Назначение
4:0	policy	Установка состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; другое - резерв/выключено
31:5	-	Резерв

8.5.6 Регистр DDR0_RESETN_PSTATUS

Регистр статуса состояния PHY DDR0.

Таблица 8.11. Формат регистра DDR0_RESETN_PSTATUS

Номер бита	Условное обозначение	Назначение
4:0	policy	Поле отражает текущую активную установку состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; другое - резерв/выключено
31:5	-	Резерв

8.5.7 Регистр DDR0_PRESETN_PPOLICY

Регистр установки состояния контроллера DDR0.

Таблица 8.12. Формат регистра DDR0_PRESETN_PPOLICY

Номер бита	Условное обозначение	Назначение
4:0	policy	Установка состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; другое - резерв/выключено
31:5	-	Резерв

8.5.8 Регистр DDR0_PRESETN_PSTATUS

Регистр статуса состояния контроллера DDR0.

Таблица 8.13. Формат регистра DDR0_RESETN_PSTATUS

Номер бита	Условное обозначение	Назначение
4:0	policy	Поле отражает текущую активную установку состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; другое - резерв/выключено
31:5	-	Резерв

8.5.9 Регистр DDR1_RESETN_PPOLICY

Регистр установки состояния PHY DDR1.

Таблица 8.14. Формат регистра DDR1_RESETN_PPOLICY

Номер бита	Условное обозначение	Назначение
4:0	policy	Установка состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; другое - резерв/выключено
31:5	-	Резерв

8.5.10 Регистр DDR1_RESETN_PSTATUS

Регистр статуса состояния PHY DDR1.

Таблица 8.15. Формат регистра DDR1_RESETN_PSTATUS

Номер бита	Условное обозначение	Назначение
4:0	policy	Поле отражает текущую активную установку состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; другое - резерв/выключено
31:5	-	Резерв

8.5.11 Регистр DDR1_PRESETN_PPOLICY

Регистр установки состояния контроллера DDR1.

Таблица 8.16. Формат регистра DDR1_PRESETN_PPOLICY

Номер бита	Условное обозначение	Назначение
4:0	policy	Установка состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; другое - резерв/выключено
31:5	-	Резерв

8.5.12 Регистр DDR1_PRESETN_PSTATUS

Регистр статуса состояния контроллера DDR1.

Таблица 8.17. Формат регистра DDR1_RESETN_PSTATUS

Номер бита	Условное обозначение	Назначение
4:0	policy	Поле отражает текущую активную установку состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; другое - резерв/выключено
31:5	-	Резерв

8.5.13 Регистр DDR0_HIF_MRR_DATA_0

DDR0 hif_mrr_data [31:0]

Формат регистра DDR0_HIF_MRR_DATA_ 0 приведен в Таблица 8.18.

Более подробное описание регистра можно найти в Разделе 24.5.1 РП.

Таблица 8.18. Формат регистра DDR0_HIF_MRR_DATA_0

Номер бита	Условное обозначение	Назначение
31:0	DDR0_HIF_MRR_DATA_0	DDR0 hif_mrr_data [31:0]

8.5.14 Регистр DDR0_HIF_MRR_DATA_1

DDR0 hif_mrr_data [63:32]

Формат регистра DDR0_HIF_MRR_DATA_1 приведен в Таблица 8.19.

Более подробное описание регистра можно найти в Разделе 24.5.1 РП.

Таблица 8.19. Формат регистра DDR0_HIF_MRR_DATA_1

Номер бита	Условное обозначение	Назначение
31:0	DDR0_HIF_MRR_DATA_1	DDR0 hif_mrr_data [63:32]

8.5.15 Регистр DDR0_HIF_MRR_DATA_2

DDR0 hif_mrr_data [95:64]

Формат регистра DDR0_HIF_MRR_DATA_ 2 приведен в Таблица 8.20.

Более подробное описание регистра можно найти в Разделе 24.5.1 РП.

Таблица 8.20. Формат регистра DDR0_HIF_MRR_DATA_2

Номер бита	Условное обозначение	Назначение
31:0	DDR0_HIF_MRR_DATA_2	DDR0 hif_mrr_data [95:64]

8.5.16 Регистр DDR0_HIF_MRR_DATA_3

DDR0 hif_mrr_data [127:96]

Формат регистра DDR0_HIF_MRR_DATA_ 3 приведен в Таблица 8.21.

Более подробное описание регистра можно найти в разделе 24.5.1 РП.

Таблица 8.21. Формат регистра DDR0_HIF_MRR_DATA_3

Номер бита	Условное обозначение	Назначение
31:0	DDR0_HIF_MRR_DATA_3	DDR0 hif_mrr_data [127:96]

8.5.17 Регистр DDR1_HIF_MRR_DATA_0

DDR1 hif_mrr_data [31:0]

Формат регистра DDR1_HIF_MRR_DATA_ 0 приведен в Таблица 8.22.

Более подробное описание регистра можно найти в Разделе 24.5.1 РП.

Таблица 8.22. Формат регистра DDR1_HIF_MRR_DATA_0

Номер бита	Условное обозначение	Назначение
31:0	DDR1_HIF_MRR_DATA_0	DDR1 hif_mrr_data [31:0]

8.5.18 Регистр DDR1_HIF_MRR_DATA_1

DDR1 hif_mrr_data [63:32]

Формат регистра DDR1_HIF_MRR_DATA_1 приведен в Таблица 8.23.

Более подробное описание регистра можно найти в Разделе 24.5.1 РП.

Таблица 8.23. Формат регистра DDR1_HIF_MRR_DATA_1

Номер бита	Условное обозначение	Назначение
31:0	DDR1_HIF_MRR_DATA_1	DDR1 hif_mrr_data [63:32]

8.5.19 Регистр DDR1_HIF_MRR_DATA_2

DDR1 hif_mrr_data [95:64]

Формат регистра DDR1_HIF_MRR_DATA_ 2 приведен в Таблица 8.24.

Более подробное описание регистра можно найти в Разделе 24.5.1 РП.

Таблица 8.24. Формат регистра DDR1_HIF_MRR_DATA_2

Номер бита	Условное обозначение	Назначение
31:0	DDR1_HIF_MRR_DATA_2	DDR1 hif_mrr_data [95:64]

8.5.20 Регистр DDR1_HIF_MRR_DATA_3

DDR1 hif_mrr_data [127:96]

Формат регистра DDR1_HIF_MRR_DATA_ 3 приведен в Таблица 8.25.

Более подробное описание регистра можно найти в Разделе 24.5.1 РП.

Таблица 8.25. Формат регистра DDR1_HIF_MRR_DATA_3

Номер бита	Условное обозначение	Назначение
31:0	DDR1_HIF_MRR_DATA_3	DDR1 hif_mrr_data [127:96]

8.5.21 Регистр DDR0_AWQOS_0

Нулевой регистр управления сигналами AWQOS AXI интерфейсов DDR0.

Формат регистра DDR0_AWQOS_0 приведен в Таблица 8.26.

Таблица 8.26. Формат регистра DDR0_AWQOS_0

Номер бита	Условное обозначение	Назначение
3:0	SDR_AWQOS	Значение сигнала AWQOS интерфейса SDR DDR0.
7:4	PCIE_AWQOS	Значение сигнала AWQOS интерфейса PCIe DDR0.
11:8	ISP_AWQOS	Значение сигнала AWQOS интерфейса ISP DDR0.
15:12	GPU_AWQOS	Значение сигнала AWQOS интерфейса GPU DDR0.
19:16	VPU_AWQOS	Значение сигнала AWQOS интерфейса VPU DDR0.
23:20	DP_AWQOS	Значение сигнала AWQOS интерфейса DP DDR0.
27:24	CPU_AWQOS	Значение сигнала AWQOS интерфейса CPU DDR0.
31:28	SERVICE_AWQOS	Значение сигнала AWQOS интерфейса Service DDR0.

8.5.22 Регистр DDR0_AWQOS_1

Первый регистр управления сигналами AWQOS AXI интерфейсов DDR0.

Формат регистра DDR0_AWQOS_1 приведен в Таблица 8.27.

Таблица 8.27. Формат регистра DDR0_AWQOS_1

Номер бита	Условное обозначение	Назначение
3:0	HSPERIPH_AWQOS	Значение сигнала AWQOS интерфейса HSPeriph DDR0.
7:4	LSPERIPH0_AWQOS	Значение сигнала AWQOS интерфейса LSPeriph0 DDR0.
11:8	LSPERIPH1_AWQOS	Значение сигнала AWQOS интерфейса LSPeriph1 DDR0.
31:12	-	Резерв

8.5.23 Регистр DDR0_ARQOS_0

Нулевой регистр управления сигналами ARQOS AXI интерфейсов DDR0.

Формат регистра DDR0_ARQOS_0 приведен в Таблица 8.28.

Таблица 8.28. Формат регистра DDR0_AWQOS_0

Номер бита	Условное обозначение	Назначение
3:0	SDR_ARQOS	Значение сигнала ARQOS интерфейса SDR DDR0.
7:4	PCIE_ARQOS	Значение сигнала ARQOS интерфейса PCIe DDR0.
11:8	ISP_ARQOS	Значение сигнала ARQOS интерфейса ISP DDR0.

Номер бита	Условное обозначение	Назначение
15:12	GPU_ARQOS	Значение сигнала ARQOS интерфейса GPU DDR0.
19:16	VPU_ARQOS	Значение сигнала ARQOS интерфейса VPU DDR0.
23:20	DP_ARQOS	Значение сигнала ARQOS интерфейса DP DDR0.
27:24	CPU_ARQOS	Значение сигнала ARQOS интерфейса CPU DDR0.
31:28	SERVICE_ARQOS	Значение сигнала ARQOS интерфейса Service DDR0.

8.5.24 Регистр DDR0_ARQOS_1

Первый регистр управления сигналами ARQOS AXI интерфейсов DDR0.

Формат регистра DDR0_ARQOS_1 приведен в Таблица 8.29.

Таблица 8.29. Формат регистра DDR0_AROS_1

Номер бита	Условное обозначение	Назначение
3:0	HSPERIPH_ARQOS	Значение сигнала ARQOS интерфейса HSPeriph DDR0.
7:4	LSPERIPH0_ARQOS	Значение сигнала ARQOS интерфейса LSPeriph0 DDR0.
11:8	LSPERIPH1_ARQOS	Значение сигнала ARQOS интерфейса LSPeriph1 DDR0.
31:12	-	Резерв

8.5.25 Регистр DDR1_ARQOS_0

Нулевой регистр управления сигналами AWQOS AXI интерфейсов DDR1.

Формат регистра DDR1_AWQOS_0 приведен в Таблица 8.30.

Таблица 8.30. Формат регистра DDR1_AWQOS_0

Номер бита	Условное обозначение	Назначение
3:0	SDR_AWQOS	Значение сигнала AWQOS интерфейса SDR DDR1.
7:4	PCIE_AWQOS	Значение сигнала AWQOS интерфейса PCIe DDR1.
11:8	ISP_AWQOS	Значение сигнала AWQOS интерфейса ISP DDR1.
15:12	GPU_AWQOS	Значение сигнала AWQOS интерфейса GPU DDR1.
19:16	VPU_AWQOS	Значение сигнала AWQOS интерфейса VPU DDR1.
23:20	DP_AWQOS	Значение сигнала AWQOS интерфейса DP DDR1.
27:24	CPU_AWQOS	Значение сигнала AWQOS интерфейса CPU DDR1.
31:28	SERVICE_AWQOS	Значение сигнала AWQOS интерфейса Service DDR1.

8.5.26 Регистр DDR1_AWQOS_1

Первый регистр управления сигналами AWQOS AXI интерфейсов DDR1.

Формат регистра DDR1_AWQOS_1 приведен в Таблица 8.31.

Таблица 8.31. Формат регистра DDR1_AWQOS_1

Номер бита	Условное обозначение	Назначение
3:0	HSPERIPH_AWQOS	Значение сигнала AWQOS интерфейса HSPeriph DDR1.
7:4	LSPERIPH0_AWQOS	Значение сигнала AWQOS интерфейса LSPeriph0 DDR1.
11:8	LSPERIPH1_AWQOS	Значение сигнала AWQOS интерфейса LSPeriph1 DDR1.

Номер бита	Условное обозначение	Назначение
31:12	-	Резерв

8.5.27 Регистр DDR1_ARQOS_0

Нулевой регистр управления сигналами ARQOS AXI интерфейсов DDR1.

Формат регистра DDR1_ARQOS_0 приведен в Таблица 8.32.

Таблица 8.32. Формат регистра DDR1_AWQOS_0

Номер бита	Условное обозначение	Назначение
3:0	SDR_ARQOS	Значение сигнала ARQOS интерфейса SDR DDR1.
7:4	PCIE_ARQOS	Значение сигнала ARQOS интерфейса PCIe DDR1.
11:8	ISP_ARQOS	Значение сигнала ARQOS интерфейса ISP DDR1.
15:12	GPU_ARQOS	Значение сигнала ARQOS интерфейса GPU DDR1.
19:16	VPU_ARQOS	Значение сигнала ARQOS интерфейса VPU DDR1.
23:20	DP_ARQOS	Значение сигнала ARQOS интерфейса DP DDR1.
27:24	CPU_ARQOS	Значение сигнала ARQOS интерфейса CPU DDR1.
31:28	SERVICE_ARQOS	Значение сигнала ARQOS интерфейса Service DDR1.

8.5.28 Регистр DDR1_ARQOS_1

Первый регистр управления сигналами ARQOS AXI интерфейсов DDR1.

Формат регистра DDR1_ARQOS_1 приведен в Таблица 8.33.

Таблица 8.33. Формат регистра DDR1_ARQOS_1

Номер бита	Условное обозначение	Назначение
3:0	HSPERIPH_ARQOS	Значение сигнала ARQOS интерфейса HSPeriph DDR1.
7:4	LSPERIPH0_ARQOS	Значение сигнала ARQOS интерфейса LSPeriph0 DDR1.
11:8	LSPERIPH1_ARQOS	Значение сигнала ARQOS интерфейса LSPeriph1 DDR1.
31:12	-	Резерв

8.5.29 Регистр CTRSECUREREGION

Регистр управления доступом к выделенным адресным диапазонам памяти.

Формат регистра CTRSECUREREGION приведен в Таблица 8.34.

Таблица 8.34. Формат регистра CTRSECUREREGION

Номер бита	Условное обозначение	Назначение
0	ENABLE	0: Доступ к выделенным адресным диапазонам разрешен. 1: Доступ к выделенным адресным диапазонам запрещен.
31:1	-	Резерв

8.5.30 Регистр BASESECUREREGION*X*_LOW

База адресного диапазона *X*. Младшая часть.

Выделенный адресный диапазон определяется маской и базой. Транзакция попадает в диапазон, если выполняется условие:

AxADDR & MaskSecureRegion*X* == BaseSecureRegion*X*.

Где MaskSecureRegion*X* = {MASKSECUREREGION*X*_HIGH, MASKSECUREREGION*X*_LOW},

и BaseSecureRegion*X* = {BASESECUREREGION*X*_HIGH, BASESECUREREGION*X*_LOW}.

Формат регистра BASESECUREREGION*X*_LOW приведен в Таблица 8.35.

Таблица 8.35. Формат регистра BASESECUREREGION*X*_LOW

Номер бита	Условное обозначение	Назначение
31:0	RANGE	База адресного диапазона *X*. Младшая часть.

8.5.31 Регистр BASESECUREREGION*X*_HIGH

База адресного диапазона *X*. Старшая часть.

Выделенный адресный диапазон определяется маской и базой. Транзакция попадает в диапазон, если выполняется условие:

AxADDR & MaskSecureRegion*X* == BaseSecureRegion*X*.

Где MaskSecureRegion*X* = {MASKSECUREREGION*X*_HIGH, MASKSECUREREGION*X*_LOW},

и BaseSecureRegion*X* = {BASESECUREREGION*X*_HIGH, BASESECUREREGION*X*_LOW}.

Формат регистра BASESECUREREGION*X*_HIGH приведен в Таблица 8.36.

Таблица 8.36. Формат регистра BASESECUREREGION*X*_LOW

Номер бита	Условное обозначение	Назначение
3:0	RANGE	База адресного диапазона *X*. Старшая часть.

8.5.32 Регистр MASKSECUREREGION*X*_LOW

Маска адресного диапазона *X*. Младшая часть.

Выделенный адресный диапазон определяется маской и базой. Транзакция попадает в диапазон, если выполняется условие:

AxADDR & MaskSecureRegion*X* == BaseSecureRegion*X*.

Где MaskSecureRegion*X* = {MASKSECUREREGION*X*_HIGH, MASKSECUREREGION*X*_LOW},

и BaseSecureRegion*X* = {BASESECUREREGION*X*_HIGH, BASESECUREREGION*X*_LOW}.

Формат регистра MASKSECUREREGION*X*_LOW приведен в Таблица 8.37.

Таблица 8.37. Формат регистра MASKSECUREREGION*X*_LOW

Номер бита	Условное обозначение	Назначение
31:0	RANGE	Маска адресного диапазона *X*. Младшая часть.

8.5.33 Регистр MASKSECUREREGION*X*_HIGH

Маска адресного диапазона *X*. Старшая часть.

Выделенный адресный диапазон определяется маской и базой. Транзакция попадает в диапазон, если выполняется условие:

AxADDR & MaskSecureRegion*X* == BaseSecureRegion*X*.

Где MaskSecureRegion*X* = {MASKSECUREREGION*X*_HIGH, MASKSECUREREGION*X*_LOW},

и BaseSecureRegion*X* = {BASESECUREREGION*X*_HIGH, BASESECUREREGION*X*_LOW}.

Формат регистра MASKSECUREREGION*X*_HIGH приведен в Таблица 8.38.

Таблица 8.38. Формат регистра MASKSECUREREGION*X*_LOW

Номер бита	Условное обозначение	Назначение
3:0	RANGE	Маска адресного диапазона *X*. Старшая часть.

8.5.34 Регистр DDR*x*_RAM_SLEEP

Регистр перевода памяти DDR*x* в режим пониженного энергопотребления.

Таблица 8.39. Формат регистра DDR*x*_RAM_SLEEP

Номер бита	Условное обозначение	Назначение
0	RAM_SLEEP	Регистр перевода write data ram и read reordering buffer в режим пониженного энергопотребления
1	CACHE_SLEEP	Регистр перевода data cache и instruction cahce в режим пониженного энергопотребления

8.5.35 Регистр GPU_BAR

Регистр приращения адреса интерфейса GPU_AXI.

Таблица 8.40. Формат регистра GPU_BAR

Номер бита	Условное обозначение	Назначение
5:0	BAR	Регистр приращения адреса интерфейса GPU AXI. Шаг приращения – 1 Гбайт. Итоговый адрес рассчитывается по формуле: AxADDR* = {BAR + AxADDR[31:30], AxADDR[29:0]}

8.5.36 Регистр HSPERPIH_BAR

Регистр приращения адреса интерфейса HSPERIPH AXI.

Таблица 8.41. Формат регистра HSPERIPH_BAR

Номер бита	Условное обозначение	Назначение
5:0	BAR	Регистр приращения адреса интерфейса HSPERPIH AXI. Шаг приращения – 1 Гбайт. Итоговый адрес рассчитывается по формуле: AxADDR* = {BAR + AxADDR[31:30], AxADDR[29:0]}

8.5.37 Регистр LSPERIPH0_BAR

Регистр приращения адреса интерфейса LSPERIPH0 AXI.

Таблица 8.42. Формат регистра LSPERIPH0_BAR

Номер бита	Условное обозначение	Назначение
5:0	BAR	Регистр приращения адреса интерфейса LSPERPIH0 AXI. Шаг приращения – 1 Гбайт. Итоговый адрес рассчитывается по формуле: AxADDR* = {BAR + AxADDR[31:30], AxADDR[29:0]}

8.5.38 Регистр LSPERIPH1_BAR

Регистр приращения адреса интерфейса LSPERIPH1 AXI

Таблица 8.43. Формат регистра LSPERIPH1_BAR

Номер бита	Условное обозначение	Назначение
5:0	BAR	Регистр приращения адреса интерфейса LSPERPIH1 AXI. Шаг приращения – 1 Гбайт. Итоговый адрес рассчитывается по формуле: AxADDR* = {BAR + AxADDR[31:30], AxADDR[29:0]}

9. ПОДСИСТЕМА HSPERIPH

9.1 Общая информация

Подсистема высокоскоростной периферии (hsperiph) включает в себя:

- EMAC0, EMAC1 – контроллеры Ethernet (до 1Гбит/с, rgmii)
- USB0,USB1– контроллеры USB3.0
- DEBUG – подсистема отладки
- NAND – контроллер флэш-памяти ONFI
- SDMMC0, SDMMC1 - контроллеры карт SD/SDIO/eMMC
- PDMA2 – DMA контроллер
- QSPI1 – контроллер SPI (Quad, Dual, Full-Duplex, Half-Duplex, XIP режимы)
- SPRAM0 – 64Кбайтная RAM подсистемы
- CTR – блок регистров, управление частотами и ресетами
- Коммутаторы SYS и DMA

Настройка и управление подсистемой осуществляется через коммутатор SYS. Основной обмен данными производится через коммутатор DMA. Практически все устройства подсистемы являются мастерами коммутатора DMA.

В подсистеме реализован контроллер PDMA2, который позволяет осуществлять пересылки данных как за пределы подсистемы, так и внутри неё. Вторым основным назначением PDMA2 является обслуживание контроллера QSPI1, который не является мастером коммутатора DMA.

Буфер SPRAM0 используется как промежуточный буфер при обменах через PDMA2 или как буфер общего назначения.

Структурная схема подсистемы показана на рисунке ниже.

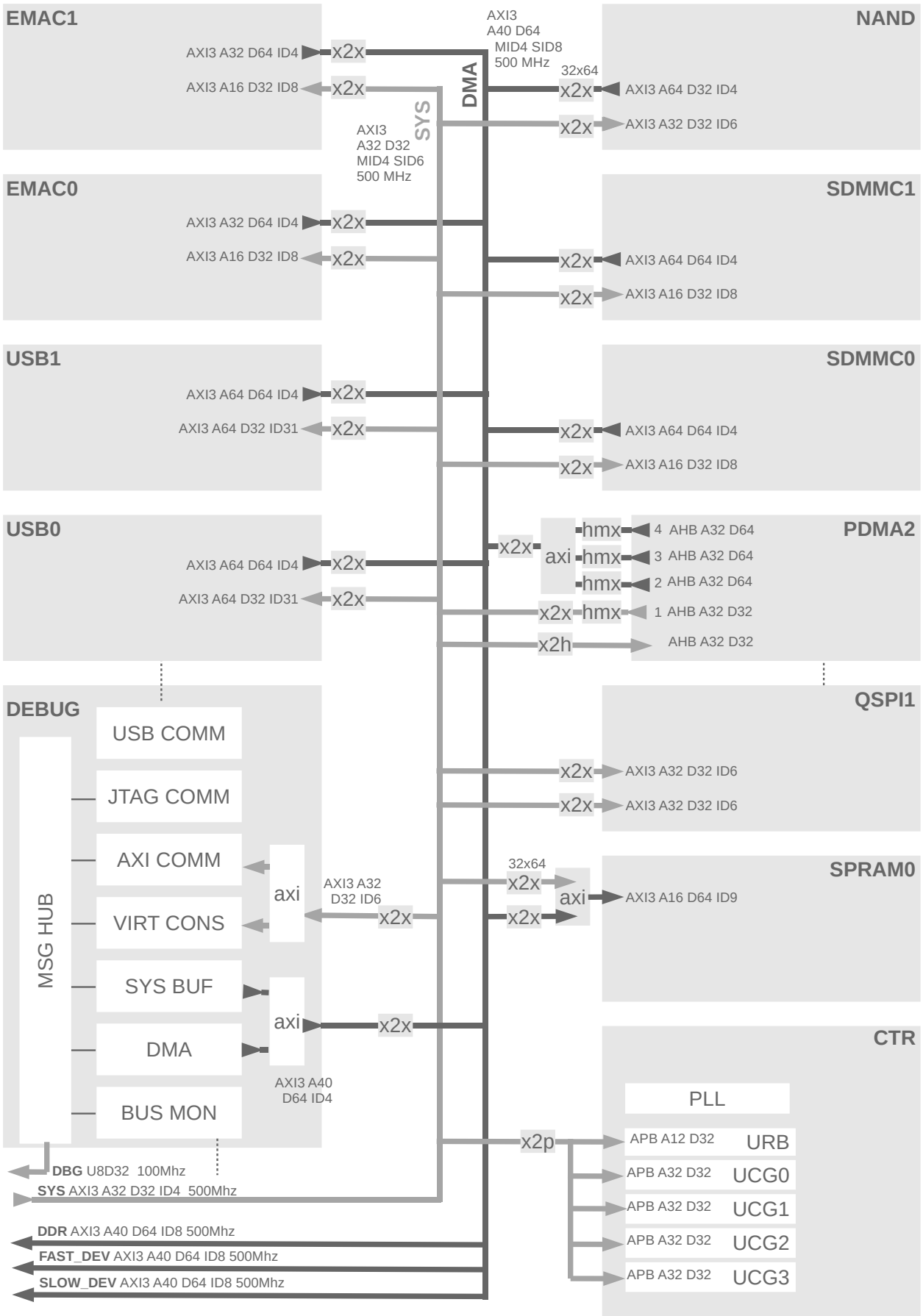


Рисунок 9.1 HSPERIPH. Структурная схема подсистемы

9.2 Синхронизация и сброс

На следующем рисунке показаны основные частоты подсистемы:

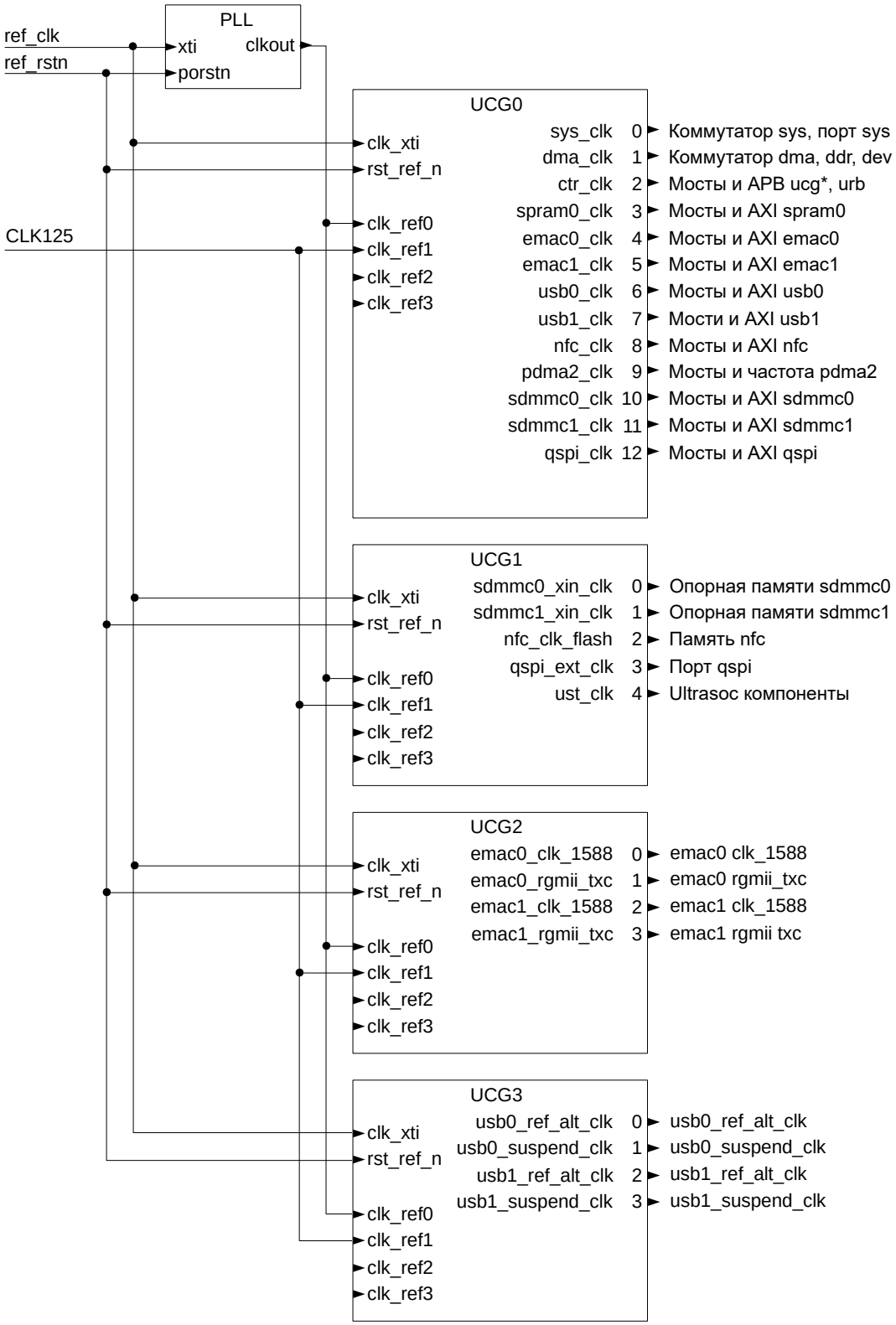


Рисунок 9.2 HSPERIPH. Схема тактирования подсистемы

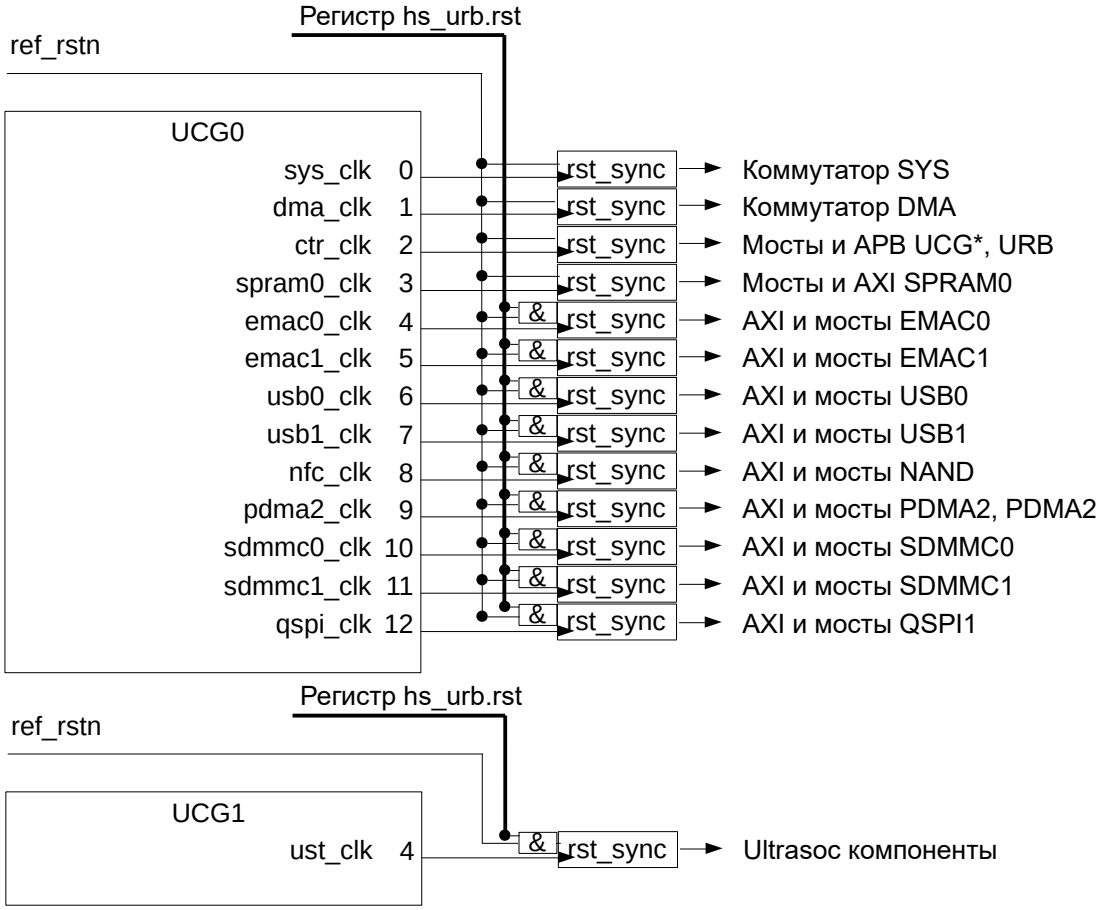


Рисунок 9.3 HSPERIPH. Схема сброса подсистемы

Таблица 9.1 Перечень основных доменов синхронизации

Домен	Источник	Частота, МГц	Описание
	ref_clk_i	27.456	Опорная частота pll
	CLK125	125	Частота 125 МГц
	ctr.pll.clkout_o	До 1000	Выходная частота pll
sys_clk	ctr.ucg0.clk_o[0]	До 500	Частота работы коммутатора SYS, порта SYS, асинхронных мостов при подключении к SYS, монитора шины SYS
	syc_clk_o		Частота порта SYS
dma_clk	ctr.ucg0.clk_o[1]	До 500	Частота работы коммутатора DMA, портов DDR, DEV, асинхронных мостов при подключении к DMA.
	ddr_dev_clk_o		Частота портов DEV, DDR
ctr_clk	ctr.ucg0.clk_o[2]	До 100	URB, блоки UCG, мосты SYS->CTR
spram0_clk	ctr.ucg0.clk_o[3]	До 500	AXI и мосты SYS,DMA SPRAM0

Домен	Источник	Частота, МГц	Описание
emac0_clk	ctr.ucg0.clk_o[4]	До 250	AXI и мосты SYS,DMA EMAC0
emac1_clk	ctr.ucg0.clk_o[5]	До 250	AXI и мосты SYS,DMA EMAC1
usb0_clk	ctr.ucg0.clk_o[6]	До 250	AXI и мосты SYS,DMA USB0
usb1_clk	ctr.ucg0.clk_o[7]	До 250	AXI и мосты SYS,DMA USB1
nfc_clk	ctr.ucg0.clk_o[8]	До 250	AXI и мосты SYS,DMA NAND
pdma2_clk	ctr.ucg0.clk_o[9]	До 500	Частота PDMA2, мостов SYS,DMA PDMA2
sdmmc0_clk	ctr.ucg0.clk_o[10]	До 200	AXI и мосты SYS,DMA SDMMC0
sdmmc1_clk	ctr.ucg0.clk_o[11]	До 200	AXI и мосты SYS,DMA SDMMC1
qspi1_clk	ctr.ucg0.clk_o[12]	До 200	AXI и мост SYS QSPI1
sdmmc0_xin_clk	ctr.ucg1.clk_o[0]	До 200	Опорная частота SDMMC0
sdmmc1_xin_clk	ctr.ucg1.clk_o[1]	До 200	Опорная частота SDMMC1
nfc_clk_flash	ctr.ucg1.clk_o[2]	До 250	Память NAND
qspi_extck	ctr.ucg1.clk_o[3]	До 200	Частота QSPI
ust_clk	ctr.ucg1.clk_o[4]	До 166	Ultrasoc компоненты
emac0_clk_1588	ctr.ucg2.clk_o[0]	До 125	EMAC0 clk_1588
emac0_rgmii_txc	ctr.ucg2.clk_o[1]	2.5, 25,125	EMAC0 rgmii_txc
emac1_clk_1588	ctr.ucg2.clk_o[2]	До 125	EMAC1 clk_1588
emac1_rgmii_txc	ctr.ucg2.clk_o[3]	2.5, 25,125	EMAC1 rgmii_txc
usb0_ref_alt_clk	ctr.ucg3.clk_o[0]	19.2,20,24,25,26,38.4,40,48,50,52,100(до 125)	Опорная PHY USB0
usb0_suspend_clk	ctr.ucg3.clk_o[1]	От 32 кГц до 125 МГц	Suspend CNTRL USB0
usb1_ref_alt_clk	ctr.ucg3.clk_o[2]	19.2,20,24,25,26,38.4,40,48,50,52,100(до 125)	Опорная PHY USB1
usb1_suspend_clk	ctr.ucg3.clk_o[3]	От 32 кГц до 125 МГц	Suspend CNTRL USB1

9.3 Прерывания

В следующей таблице приведены прерывания от HSPERIPH

Таблица 9.2 Список прерываний HSPERIPH

Сигнал	Комментарий
emac0_axi_intr	см. документацию на блок

Сигнал	Комментарий
emac0_axi_wkuprcvd	см. документацию на блок
emac1_axi_intr	см. документацию на блок
emac1_axi_wkuprcvd	см. документацию на блок
sdmmc0_axi_intr	см. документацию на блок
sdmmc0_axi_wkup	см. документацию на блок
sdmmc1_axi_intr	см. документацию на блок
sdmmc1_axi_wkup	см. документацию на блок
nfc_int_to_arm	см. документацию на блок
pdma2_int_combined	см. документацию на блок
qspi_intreq	см. документацию на блок
dbg_irq_comb	см. документацию на блок
usb0_irq_comb	см. документацию на блок
usb1_irq_comb	см. документацию на блок

9.4 Контактные площадки

В подсистеме HSPERIPH для всех блоков кроме USB используются контактные площадки (КП) типа LVCMOS. LVCMOS КП работают на напряжениях 1.8 В или 3.3 В. Выбор рабочего напряжения осуществляется через регистры настроек контактных площадок, а для блоков SDMMC через регистры SDMMC. По сбросу все LVCMOS КП находятся в 3-ем состоянии с рабочим напряжением 3.3 В.

Все КП распределены по группам с незавизимым питанием, группы КП с независимым питаний показаны на следующем рисунке:

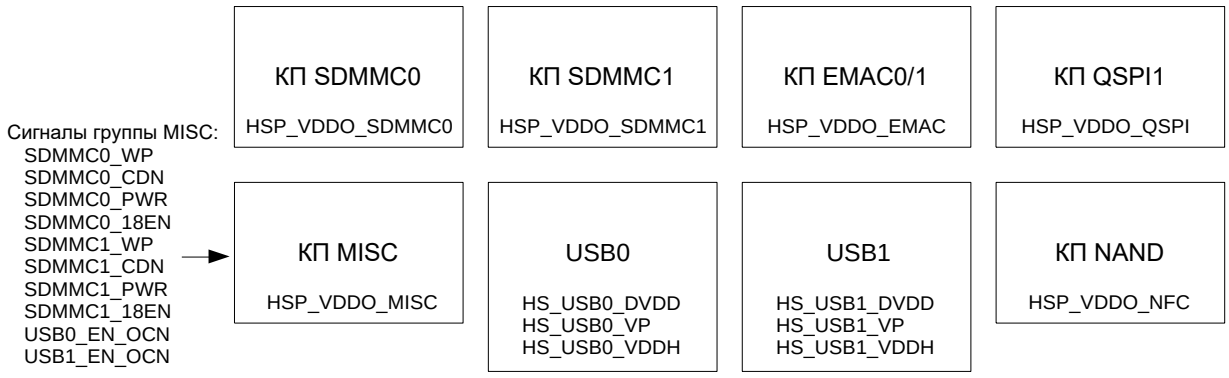


Рисунок 9.4 HSPERIPH. Домены питания подсистемы

Контактные площадки группы NAND могут использоваться контролером NAND или контроллером GNSS интегрированным в подсистеме DDR. Сделано это для расширения возможностей применения GNSS – если внешний RF тракт работает на напряжении 1.8, то GNSS использует свои контактные площадки интегрированные в подсистеме SDR, если внешний RF тракт работает на напряжении 3.3, то GNSS использует контактные площадки интегрированные в подсистеме HSPERIPH.

Внимание! Одновременная работа блока GNSS с КП NAND и контроллера NAND невозможна.

На следующем рисунке схематично показано разделение КП между блоками GNSS и NAND:

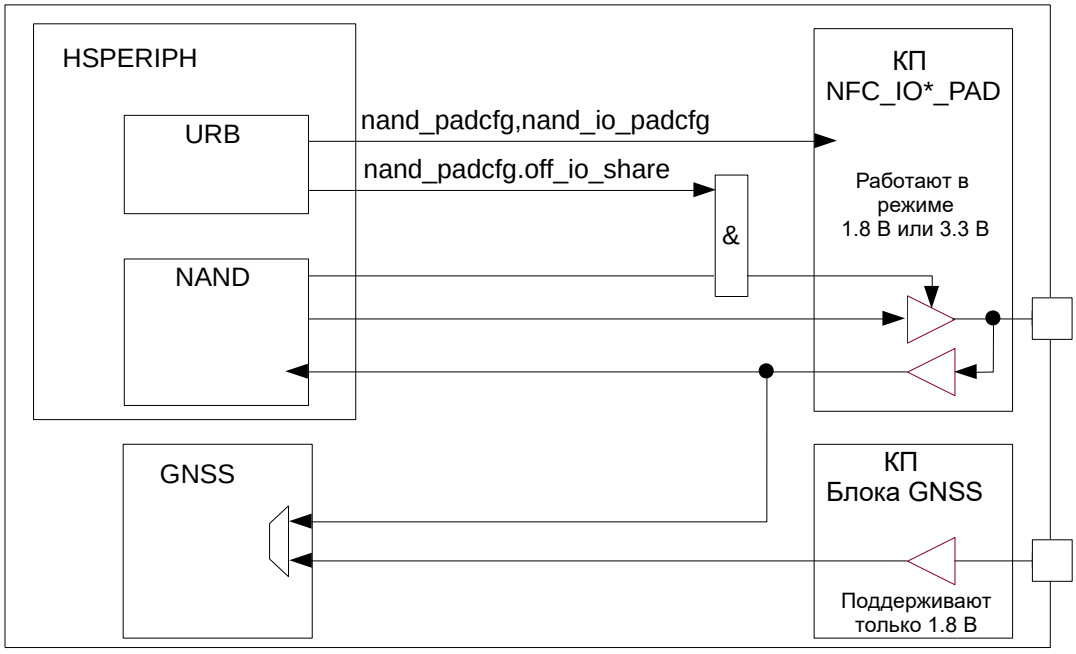


Рисунок 9.5 Разделение КП между NAND и блоком GNSS

9.4.1 Настройки КП

Для настройки КП используются регистры URB.*_padcfg, которые содержат следующие поля:

- EN. Включение КП. Позволяет переводить КП в Z-состояние независимо от управляющих сигналов контроллеров интерфейсов подключенных к КП.
    - 0 – Все выходы в 3-м состоянии. Приём данных запрещён.
    - 1 – Разрешен активный уровень на выходах, разрешён приём данных.

- V18. Настройка режима КП – 1.8 В или 3.3 В. Обязательно должна соответствовать рабочему напряжению.
- CLE. Тестовая настройка.
- PU/PD/SUS. Подтяжка КП – pull-up/pull-down/sustain. Типичное значение подтяжки – 30 кОм, в зависимости от PVT-угла меняется в диапазоне от 16 до 43 кОм. В один момент времени только один из этих сигналов может принимать единичное значение. Подтяжка действует только в режиме приемника.
- SL[1:0] (Slew Rate Limit). Ограничение скорости нарастания сигнала КП.
    - SL=2’b00 – наименьшая скорость нарастания
    - SL=2’b11 – наибольшая скорость
- CTL[5:0]. Максимальный ток КП
    - 6'b000000 - выход находится в 3-ем состоянии.
    - 6'b000001 – 2 мА
    - 6'b000011 – 4 мА
    - 6'b000111 – 6 мА
    - 6'b001111 – 8 мА
    - 6'b011111 – 10 мА
    - 6'b111111 – 12 мА
- OD. Включение режима open-drain. '1' - режим open-drain включен.
- ST. Включение триггера Шмитта для входных КП. '1' - триггер включен. Не влияет на КП, настроенные как выходы.

9.4.2 Настройка питания КП

Переключение питания КП должно производиться с особой осторожностью и должно соответствовать настройке КП. Далее описаны различные возможные комбинации настроек КП и напряжения питания.

Извне на КП подано напряжение 3.3В, а КП настроена на 1.8В.

Такая ситуация недопустима и может привести к необратимым повреждениям КП.

Извне на КП подано напряжение 1.8В, а КП настроена на 3.3В.

Повреждения КП не будет, но при этом не гарантируется нормальная работа КП (либо не будет работать, либо только в некоторых PVT-углах на пониженных частотах)

Извне на КП подано напряжение 3.3В, КП настроена на 3.3В.

Нормальный режим. При переключении в режим 3.3В из режима 1.8В необходимо соблюдать следующую последовательность шагов (VDDO-внешнее напряжение, V18-настройка КП):

1. Начальное состояние – VDDO=1.8В, V18=1.
2. Изменение настройки КП. VDDO=1.8В, V18=0
3. Постепенное переключение напряжения с 1.8В до 3.3В за время ~15мкс (минимум). VDDO=3.3В, V18=0.
4. Ожидание ~2мкс.
5. Нормальная работа с КП.

Извне на КП подано напряжение 1.8В, КП настроена на 1.8В.

Нормальный режим. При переключении в режим 1.8В из режима 3.3В необходимо соблюдать следующую последовательность шагов (VDDO-внешнее напряжение, V18-настройка КП):

1. Начальное состояние – VDDO=3.3В, V18=0.
2. Переключение напряжения КП до стабильного уровня 1.8В (VDDO=1.8В, V18=0)
3. Изменение настройки КП. VDDO=3.3В, V18=1.
4. Ожидание ~2мкс.
5. Нормальная работа с КП.

9.4.3 Ограничение MISC_VDDO

Внешний сигнал питания MISC_VDDO используется не только для КП группы MISC но и для КП питания PLL HSP_VDDA_PLL. Считается, что по сбросу MISC_VDDO равно 3,3 В. Не рекомендуется менять настройки и переключать питание MISC_VDDO во время работы с подсистемной HSPERIPH.



Рисунок 9.6 HSPERIPH. MISC_VDDO запитывает КП HSP_VDDA_PLL

9.5 Коммутаторы DMA и SYS

9.5.1 Параметры портов DDR, DEV, SYS

В следующих таблицах приведены параметры DDR, DEV, SYS.

Таблица 9.3 Параметры портов DDR и DEV

Параметр	Значение
Тип	AXI3
Роль	Мастер
Разрядность адреса	40
Разрядность данных	64
Разрядность AXI ID	8

Таблица 9.4 Параметры порта SYS

Параметр	Значение
Тип	AXI3
Роль	Слейв
Разрядность адреса	32
Разрядность данных	32
Разрядность AXI ID	4

9.5.2 Сигналы AxPROT, AxCACHE, AxLOCK

Таблица 9.5 Управление сигналами AxPROT, AxCACHE, AxLOCK коммутатора DMA

Блок	AxPROT	AxCACHE	AxLOCK
EMAC0/1	3’b010	См. регистр dma_axcache	2’b00
USB0/1	3’b010	Настраивается в контроллере	2’b00

Блок	AxPROT	AxCACHE	AxLOCK
DEBUG.DMA	3’b010	См. регистр dma_axcache	Настраивается в контроллере
DEBUG.SYS BUF	3’b010	См. регистр dma_axcache	2’b00
NAND	3’b010	См. регистр dma_axcache	2’b00
SDMMC0/1	3’b010	См. регистр dma_axcache	2’b00
PDMA2	Настраивается в PDMA2. AxPROT[2]= ~HPROT[0]; AxPROT[1] = 1; AxPROT[0]= HPROT[1];	См. регистр dma_axcache	Настраивается в PDMA2. AxLOCK[0]=1’b0; AxLOCK[1]=HLOCK;

9.5.3 Матрицы коммутации

В таблицах ниже описаны матрицы коммутации SYS и DMA коммутаторов.

Таблица 9.6 Доступность портов и устройств для мастеров коммутатора DMA

	DDR	FAST_DEV	SLOW_DEV	SPRAM0
EMAC0	+	+	+	+
EMAC1	+	+	+	+
USB0	+	+	+	+
USB1	+	+	+	+
NAND	+	+	+	+
SDMMC0	+	+	+	+
SDMMC1	+	+	+	+
PDMA2 (2,3,4)	+	+	+	+
DEBUG	+	+	+	+

Таблица 9.7 Доступность портов и устройств для мастеров коммутатора SYS

	EMAC0	EMAC1	USB0	USB1	NAND	SDMMC0	SDMMC1	PDMA2	QSPI1	QSPI1 XIP	DEBUG	SPRAM0	URB, UCG*
Порт SYS	+	+	+	+	+	+	+	+	+	+	+	+	+

	EMAC0	EMAC1	USB0	USB1	NAND	SDMMC0	SDMMC1	PDMA2	QSPI1	QSPI1 XIP	DEBUG	SPRAM0	URB, UCG*
PDMA2 (порт 1)	+	+	+	+	+	+	+	+	+	+	-	+	-

9.5.4 Приоритеты

Приоритеты мастеров DMA (у меньшего значения больший приоритет):

1 - emac0
2 - emac1
3 - sdmmc0
4 - sdmmc1
5 - nfc
6 - usb0
7 - usb1
8 - pdma2
9 - dbg

Приоритеты мастеров SYS (у меньшего значения больший приоритет):

1 - sys
2 - pdma2

9.5.5 Преобразование адреса DDR

При обращениях от EMAC0, EMAC1, PDMA2 через порт DDR адрес обращения преобразуется снаружи подсистемы:

{DDR.URB.hsperiph_bar + hsperiph_DDR_ADDR[31:30], hsperiph_DDR_ADDR[29:0]}

Адрес DDR необходимо правильно настроить как в регистрах *bar HSPERIPH.URB, так и в регистрах DDR.URB.

9.6 Карта памяти

Таблица 9.8 Диапазоны адресов коммутатора DMA

Диапазон	Порт приёмник
'h00_0000_0000-'h00_0000_FFFF	SPRAM0
'h00_0001_0000-'h00_02FF_FFFF	SLOW_DEV
'h00_0300_0000-'h00_03FF_FFFF	FAST_DEV

Диапазон	Порт приёмник
'h00_0400_0000-'h00_5FFF_FFFF	SLOW_DEV
'h00_6000_0000-'h00_7FFF_FFFF	FAST_DEV
'h00_8000_0000-'h00_FFFF_FFFF	DDR
'h04_0000_0000-'h07_7FFF_FFFF	FAST_DEV
'h07_8000_0000-'h0F_FFFF_FFFF	DDR
'h80_0000_0000-'hFF_FFFF_FFFF	FAST_DEV

Таблица 9.9 Диапазоны адресов коммутатора SYS

Диапазон	Используется	Порт-приёмник	№ Слейва
‘h0000_0000-’h0000_FFFF	64 Кбайта	SPRAM0	11
‘h1000_0000-’h100F_FFFF	1 Мбайт	USB0	4
‘h1010_0000-’h101F_FFFF	1 Мбайт	USB1	5
‘h1020_0000-’h1020_FFFF	4 Кбайта	EMAC0	2
‘h1021_0000-’h1021_FFFF	4 Кбайта	EMAC1	3
‘h1022_0000-’h1022_FFFF	256 Байт	SDMMC0	6
‘h1023_0000-’h1023_FFFF	256 Байт	SDMMC1	7
‘h1024_0000-’h1024_FFFF	256 Байт	NAND	8
‘h1025_0000-’h1025_FFFF	1 Кбайт	PDMA2	9
‘h1026_0000-’h1026_FFFF	128 Байт	QSPI1	10
‘h1040_0000-’h1040_FFFF	512 Байт	URB	1
‘h1041_0000-’h1041_FFFF	64 Байта	UCG0	1
‘h1042_0000-’h1042_FFFF	64 Байта	UCG1	1
‘h1043_0000-’h1043_FFFF	64 Байта	UCG2	1
‘h1044_0000-’h1044_FFFF	64 Байта	UCG3	1
‘h1045_0000-’h1045_FFFF	4 Кбайта	DEBUG. Virtual Console	12
‘h1046_0000-’h1046_FFFF	64 Байта	DEBUG. AXI Communicator	12
‘h5000_0000-’h5FFF_FBF8	~32 Мбайта	QSPI1 XIP	13

При работе с адресным пространством подсистемы hsperiph не рекомендуется обращаться в зарезервированные или неопределенные области памяти. Обращение в такие области может привести к непредсказуемым последствиям.

9.7 Регистры URB

Перечень регистров приведен в Таблица 9.10.

Таблица 9.10. Перечень программно-доступных регистров hsperiph urb

Смещение	Условное обозначение	Название регистра	Исходное состояние	Тип доступа
0x000	pll_cfg	Регистр конфигурации pll	00000000	RW
0x004	pll_diag	Регистр диагностики pll	00000002	RW
0x008	rst	Регистр сброса компонентов	000003FF	RW
0x00C	refclk	Регистр настройки опорных частот UCG	00000050	RW
0x010	qspi1_xip_en_req	Регистр включения режима XIP QSPI1	00000000	RW
0x014	qspi1_xip_en_out	Регистр статуса режима XIP QSPI1	00000000	RO
0x018	qspi1_xip_cfg	Регистр конфигурации режима XIP QSPI1	20010003	RW
0x01C	qspi1_padcfg	Общий регистр настройки КП QSPI1	00000004	RW
0x020	qspi1_ss_padcfg	Дополнительный регистр настройки КП QSPI1_SS (0, 1, 2, 3)	00000078	RW
0x024	qspi1_siso_padcfg	Дополнительный регистр настройки КП QSPI1_SISO (0, 1, 2, 3)	00000078	RW
0x028	qspi1_sclk_padcfg	Дополнительный регистр настройки КП QSPI1_SCLK	00000078	RW
0x02C	sdmmc0_padcfg	Регистр настройки режима работы КП SDMMC0 кроме КП SDMMC0_18EN и SDMMC0_PWR	00186A00	RW
0x030	sdmmc0_clk_padcfg	Дополнительный регистр настройки КП SDMMC0_CLK	00000078	RW
0x034	sdmmc0_cmd_padcfg	Дополнительный регистр настройки КП SDMMC0_CMD	00000078	RW
0x038	sdmmc0_dat_padcfg	Дополнительный регистр настройки КП SDMMC0_DAT (0 – 7)	00000078	RW
0x03C	sdmmc0_corecfg_0	Регистр 0 настроек SDMMC0	01000077	RW
0x040	sdmmc0_corecfg_1	Регистр 1 настроек SDMMC0	35EE1980	RW
0x044	sdmmc0_corecfg_2	Регистр 2 настроек SDMMC0	00000000	RW
0x048	sdmmc0_corecfg_3	Регистр 3 настроек SDMMC0	00000000	RW
0x04C	sdmmc0_corecfg_4	Регистр 4 настроек SDMMC0	00000000	RW
0x050	sdmmc0_corecfg_5	Регистр 5 настроек SDMMC0	00000000	RW
0x054	sdmmc0_corecfg_6	Регистр 6 настроек SDMMC0	00000000	RW
0x058	sdmmc0_corecfg_7	Регистр 7 настроек SDMMC0	000C0000	RW
0x05C	sdmmc0_sdhc_dbg0	Регистр 0 отладки SDMMC0	00000000	RO
0x060	sdmmc0_sdhc_dbg1	Регистр 1 отладки SDMMC0	00000000	RO
0x064	sdmmc0_sdhc_dbg2	Регистр 2 отладки SDMMC0	00000000	RO
0x068	sdmmc1_padcfg	Регистр настройки режима работы КП SDMMC1 кроме КП SDMMC0_18EN и SDMMC1_PWR	00186A00	RW
0x06C	sdmmc1_clk_padcfg	Дополнительный регистр настройки КП SDMMC1_CLK	00000078	RW
0x070	sdmmc1_cmd_padcfg	Дополнительный регистр настройки КП SDMMC1_CMD	00000078	RW
0x074	sdmmc1_dat_padcfg	Дополнительный регистр настройки КП SDMMC1_DAT (0 – 7)	00000078	RW
0x078	sdmmc1_corecfg_0	Регистр 0 настроек SDMMC0	01000077	RW
0x07C	sdmmc1_corecfg_1	Регистр 1 настроек SDMMC1	35EE1980	RW
0x080	sdmmc1_corecfg_2	Регистр 2 настроек SDMMC2	00000000	RW
0x084	sdmmc1_corecfg_3	Регистр 3 настроек SDMMC3	00000000	RW
0x088	sdmmc1_corecfg_4	Регистр 4 настроек SDMMC4	00000000	RW
0x08C	sdmmc1_corecfg_5	Регистр 5 настроек SDMMC5	00000000	RW

0x090	sdmmc1_corecfg_6	Регистр 6 настроек SDMMC6	00000000	RW
0x094	sdmmc1_corecfg_7	Регистр 7 настроек SDMMC7	00040000	RW
0x098	sdmmc1_sdhc_dbg0	Регистр 0 отладки SDMMC0	00000000	RO
0x09C	sdmmc1_sdhc_dbg1	Регистр 1 отладки SDMMC1	00000000	RO
0x0A0	sdmmc1_sdhc_dbg2	Регистр 2 отладки SDMMC2	00000000	RO
0x0A4	pdma2_dbg0	Регистр pdma2 dbg 0	00000000	RO
0x0A8	pdma2_dbg1	Регистр pdma2 dbg 1	00000000	RO
0x0AC	pdma2_dbg2	Регистр pdma2 dbg 2	00000000	RO
0x0B0	pdma2_dbg3	Регистр pdma2 dbg 3	00000000	RO
0x0B4	pdma2_dbg4	Регистр pdma2 dbg 4	00000000	RO
0x0B8	pdma2_dbg5	Регистр pdma2 dbg 5	00000000	RO
0x0BC	pdma2_dbg6	Регистр pdma2 dbg 6	00000000	RO
0x0C0	pdma2_dbg7	Регистр pdma2 dbg 7	00000000	RO
0x0C4	pdma2_dbg8	Регистр pdma2 dbg 8	00000000	RO
0x0C8	pdma2_bar0	Базовый адрес 0 axi PDMA2	00000000	RW
0x0CC	pdma2_bar1	Базовый адрес 1 axi PDMA2	00000000	RW
0x0D0	pdma2_bar2	Базовый адрес 2 axi PDMA2	00000000	RW
0x0D4	pdma2_hmx_ctr	Регистр управления сигналами hmx	00F00000	RW
0x0D8	usb0_cntr_misc	Управление контроллером USB0	0000001F	RW
0x0DC	usb0_phy_ctr	Регистр управления phy USB0	02C80000	RW
0x0E0	usb0_phy_params0	Параметры phy USB0 0	48CF0D0C	RW
0x0E4	usb0_phy_params1	Параметы phy USB0 1	9FDC7000	RW
0x0E8	usb0_phy_cr	Управляющий регистр phy USB0	00000000	RW
0x0EC	usb0_phy_tst	Регистр тестовых сигналов phy USB0	00000000	RW
0x0F0	usb0_phy_tst_bypass	Тест bypass phy USB0	00000000	RW
0x0F4	usb0_phy_rtune	Подстройка rtune USB0	00000000	RW
0x0F8	usb0_dbg0	Регистр отладки USB 0	00000000	RO
0x0FC	usb0_dbg1	Регистр отладки USB 1	00000000	RO
0x100	usb0_dbg2	Регистр отладки USB 2	00000000	RO
0x104	usb0_dbg3	Регистр отладки USB 3	00000000	RO
0x108	usb0_dbg4	Регистр отладки USB 4	00000000	RO
0x10C	usb1_cntr_misc	Управление контроллером USB1	0000001F	RW
0x110	usb1_phy_ctr	Регистр управления phy USB1	02C80000	RW
0x114	usb1_phy_params0	Параметры phy USB1 0	48CF0D0C	RW
0x118	usb1_phy_params1	Параметры phy USB1 1	9FDC7000	RW
0x11C	usb1_phy_cr	Управляющий регистр phy USB1	00000000	RW
0x120	usb1_phy_tst	Регистр тестовых сигналов phy USB1	00000000	RW
0x124	usb1_phy_tst_bypass	Тест bypass phy USB1	00000000	RW
0x128	usb1_phy_rtune	Подстройка rtune USB1	00000000	RW
0x12C	usb1_dbg0	Регистр отладки USB 0	00000000	RO
0x130	usb1_dbg1	Регистр отладки USB 1	00000000	RO
0x134	usb1_dbg2	Регистр отладки USB 2	00000000	RO
0x138	usb1_dbg3	Регистр отладки USB 3	00000000	RO
0x13C	usb1_dbg4	Регистр отладки USB 4	00000000	RO
0x140	emac_padcfg	Общий регистр настройки режима работы КП обоих контроллеров EMAC0 и EMAC1.	00000000	RW
0x144	emac0_padcfg	Дополнительный регистр настройки КП EMAC0	00000000	RW
0x148	emac0_tx_padcfg	Дополнительный регистр настройки КП EMAC0_RGMII_TXD*, EMAC0_RGMII_TXCTL	00000078	RW
0x14C	emac0_txc_padcfg	Дополнительный регистр настройки КП EMAC0_RGMII_TXC	00000078	RW
0x150	emac0_rx_padcfg	Дополнительный регистр настройки КП EMAC0_RGMII_RXD*, EMAC0_RGMII_RXCTL	00000000	RW

0x154	emac0_rxc_padcfg	Дополнительный регистр настройки КП EMAC0_RGMII_RXC	00000000	RW
0x158	emac0_md_padcfg	Дополнительный регистр настройки КП EMAC0_RGMII_MDIO	00000078	RW
0x15C	emac0_mdc_padcfg	Дополнительный регистр настройки КП EMAC0_RGMII_MDC	00000078	RW
0x160	emac0_bar	Базовый адрес axi EMAC0	00000000	RW
0x164	emac1_padcfg	Дополнительный регистр настройки КП EMAC1	00000000	RW
0x168	emac1_tx_padcfg	Дополнительный регистр настройки КП EMAC1_RGMII_TXD*, EMAC1_RGMII_TXCTL	00000078	RW
0x16C	emac1_txc_padcfg	Дополнительный регистр настройки КП EMAC1_RGMII_TXC	00000078	RW
0x170	emac1_rx_padcfg	Дополнительный регистр настройки КП EMAC1_RGMII_RXD*, EMAC1_RGMII_RXCTL	00000000	RW
0x174	emac1_rxc_padcfg	Дополнительный регистр настройки КП EMAC1_RGMII_RXC	00000000	RW
0x178	emac1_md_padcfg	Дополнительный регистр настройки КП EMAC1_RGMII_MDIO	00000078	RW
0x17C	emac1_mdc_padcfg	Дополнительный регистр настройки КП EMAC1_RGMII_MDC	00000078	RW
0x180	emac1_bar	Базовый адрес axi EMAC1	00000000	RW
0x184	nand_padcfg	Общий регистр настройки КП NAND	00000014	RW
0x188	nand_ale_cle_padcfg	Дополнительный регистр настройки КП NFC_ALE, NFC_CLE	00000078	RW
0x18C	nand_cen_padcfg	Дополнительный регистр настройки КП NFC_CEN*	00000078	RW
0x190	nand_ren_padcfg	Дополнительный регистр настройки КП NFC_REN	00000078	RW
0x194	nand_wen_padcfg	Дополнительный регистр настройки КП NFC_WEN	00000078	RW
0x198	nand_io_padcfg	Дополнительный регистр настройки КП NFC_IO*	00000078	RW
0x19C	nand_dqs_padcfg	Дополнительный регистр настройки КП NFC_DQS	00000078	RW
0x1A0	nand_rbn_padcfg	Дополнительный регистр настройки КП NFC_RBN*	00000078	RW
0x1A4	misc_padcfg	Регистр настройки КП MISC	0000C843	RW
0x1A8	dbg_ctr	Регистр отладки	00000000	RW
0x1AC	dma_axcache	Регистр настройки сигналов AXCACHE DMA	00000000	RW

9.7.1 pll_cfg

Регистр конфигурации pll.

Формат регистра pll_cfg приведен в следующей таблице:

Таблица 9.11. Поля регистра pll_cfg

Номер бита	Поле	Описание	Сброс
31	lock	Бит указывает окончание настройки PLL на новую заданную частоту работы. При начале процедуры настройки, либо при выключенной PLL сбрасывается в 0	0
30:27	nr_man	Коэффициент nr настройки pll	0

Номер бита	Поле	Описание	Сброс
26:14	nf_man	Коэффициент nf настройки pll	0
13:10	od_man	Коэффициент od настройки pll	0
9	man	При бите заданном в 1 и sel > 0 конфигурация pll определяется полям *_man данного регистра. Выходная частота определяется по формуле ref_clk * (nf_man + 1) / (nr_man + 1) / (od_man + 1). Поля *_man рекомендуется задавать исходя из значений генерируемым скриптом настройки pll. Значение выходной частоты не должно превышать 3600 МГц.	0
8	-	Резерв	0
7:0	sel	Определяет выходную частоту и режим работы PLL: 0x0 - опорная частота ref_clk, PLL выключена; 0x1 - ref_clk * 2; 0x2 - ref_clk * 3; 0x3 - ref_clk * 4; ... - ...; 0x73 - ref_clk * 116; 0x74 - ref_clk * 116; ... - ...; 0xFF - ref_clk * 116; При sel > 0 и man = 1, конфигурация pll определяется полями *_man данного регистра	0

9.7.2 pll_diag

Регистр диагностики pll.

Формат регистра pll_diag приведен в следующей таблице:

Таблица 9.12. Поля регистра pll_diag

Номер бита	Поле	Описание	Сброс
31:5	-	Резерв	0
4	fbslip	Диагностический выход pll - регистрирует ситуацию feedback cycle slip	0
3	rfslip	Диагностический выход pll - регистрирует ситуацию referency cycle slip	0
2	fasten	Диагностический вход pll - разрешает механизм fast lock	0
1	ensat	Диагностический вход pll - разрешает насыщение, в нормальном режиме - 1	1
0	test	Диагностический вход pll - bypass опорной частоты pll на выход при 1	0

9.7.3 rst

Регистр сброса компонентов.

Формат регистра rst приведен в следующей таблице:

Таблица 9.13. Поля регистра rst

Номер бита	Поле	Описание	Сброс
31:26	-	Резерв	0

Номер бита	Поле	Описание	Сброс
25	emac1_wr	Необходимо установить в единцу при записи в соответствующий бит сброса. Автоматически сбрасывается в 0.	0
24	emac0_wr	Необходимо установить в единцу при записи в соответствующий бит сброса. Автоматически сбрасывается в 0.	0
23	usb1_wr	Необходимо установить в единцу при записи в соответствующий бит сброса. Автоматически сбрасывается в 0.	0
22	usb0_wr	Необходимо установить в единцу при записи в соответствующий бит сброса. Автоматически сбрасывается в 0.	0
21	sdmmc1_wr	Необходимо установить в единцу при записи в соответствующий бит сброса. Автоматически сбрасывается в 0.	0
20	sdmmc0_wr	Необходимо установить в единцу при записи в соответствующий бит сброса. Автоматически сбрасывается в 0.	0
19	nfc_wr	Необходимо установить в единцу при записи в соответствующий бит сброса. Автоматически сбрасывается в 0.	0
18	qspi_wr	Необходимо установить в единцу при записи в соответствующий бит сброса. Автоматически сбрасывается в 0.	0
17	pdma2_wr	Необходимо установить в единцу при записи в соответствующий бит сброса. Автоматически сбрасывается в 0.	0
16	dbg_wr	Необходимо установить в единцу при записи в соответствующий бит сброса. Автоматически сбрасывается в 0.	0
15:10	-	Резерв	0
9	emac1	Запись в данный бит проходит только при установленной маске записи - emac1_wr.	1
8	emac0	Запись в данный бит проходит только при установленной маске записи - emac0_wr.	1
7	usb1	Запись в данный бит проходит только при установленной маске записи - usb1_wr.	1
6	usb0	Запись в данный бит проходит только при установленной маске записи - usb0_wr.	1
5	sdmmc1	Запись в данный бит проходит только при установленной маске записи - sdmmc1_wr.	1
4	sdmmc0	Запись в данный бит проходит только при установленной маске записи - sdmmc0_wr.	1
3	nfc	Запись в данный бит проходит только при установленной маске записи - nfc_wr.	1
2	qspi	Запись в данный бит проходит только при установленной маске записи - qspi_wr.	1
1	pdma2	Запись в данный бит проходит только при установленной маске записи - pdma2_wr.	1
0	dbg	Запись в данный бит проходит только при установленной маске записи - dbg_wr.	1

9.7.4 refclk

Регистр настройки опорных частот UCG.

Формат регистра refclk приведен в следующей таблице:

Таблица 9.14. Поля регистра refclk

Номер бита	Поле	Описание	Сброс
31:8	-	Резерв	0
7:6	ucg3	Выбор опорной частоты UCG3 (управляет входом sel_clk_ref_i UCG).	1
5:4	ucg2	Выбор опорной частоты UCG2 (управляет входом sel_clk_ref_i UCG).	1
3:2	ucg1	Выбор опорной частоты UCG1 (управляет входом sel_clk_ref_i UCG).	0
1:0	ucg0	Выбор опорной частоты UCG0 (управляет входом sel_clk_ref_i UCG).	0

9.7.5 qspi1_xip_en_req

Регистр включения режима XIP QSPI1.

Формат регистра qspi1_xip_en_req приведен в следующей таблице:

Таблица 9.15. Поля регистра qspi1_xip_en_req

Номер бита	Поле	Описание	Сброс
31:1	-	Резерв	0
0	en	Для включения режим XIP установить бит в 1, обращения по адресам памяти QSPI0 и изменение регистра XIP_CFG запрещены до установки регистра XIP_OUT_EN в 1. Для отключения режима XIP установить бит в 0, обращения по адресам памяти QSPI0 и изменение регистра XIP_CFG запрещены до установки регистра XIP_OUT_EN в 0	0

9.7.6 qspi1_xip_en_out

Регистр статуса режима XIP QSPI1.

Формат регистра qspi1_xip_en_out приведен в следующей таблице:

Таблица 9.16. Поля регистра qspi1_xip_en_out

Номер бита	Поле	Описание	Сброс
31:1	-	Резерв	0
0	status	Статус режима XIP, 0 - режим выключен, 1 - режим включен;	0

9.7.7 qspi1_xip_cfg

Регистр конфигурации режима XIP QSPI1.

Формат регистра qspi1_xip_cfg приведен в следующей таблице:

Таблица 9.17. Поля регистра qspi1_xip_cfg

Номер бита	Поле	Описание	Сброс
31	-	Резерв	0

Номер бита	Поле	Описание	Сброс
30	hpen	0 - режим High Performance отключен, 1 - режим High Performance включен	0
29	le32	0 - читаемые данные Big Endian, 1 - читаемы данные 32-битные Little Endian	1
28	addr4	4-байтный режим адресации. 0 - контроллер выдает 3 байта адреса, 1 - контроллер выдает 4 байта адреса.	0
27	cpol	Режим CPOL	0
26	cpha	Режим CPHA	0
25:24	hp_end_dummy	Задает количество тактов ожидания, необходимое для выхода из режима High Performance mode. Используется контроллером при выходе из XIP режима, чтобы убедиться что Flash память не осталась в режиме HP. Для режима Dual количество тактов ожидания равно данному значению умноженному на 4. Для режима Quad количество тактов ожидания равно данному значению умноженному на 2.	0
23:20	dummy_cycles	Количество тактов ожидания при приеме. Для режимов Fast Read, Fast Read Dual Output, Fast Read Quad Output количество тактов ожидания равно данному значению умноженному на 8. Для режима Fast Read Dual I/O количество тактов ожидания равно данному значению умноженному на 4 плюс 4. Для режима Fast Read Quad I/O количество тактов ожидания равно данному значению умноженному на 2 плюс 2.	0
19:16	ssen	Выбор активного SS QSPI0 4'b0001 - Slave Select 0, 8'b0010 - Slave Select 1, ... 8'b1000 - Slave Select 3. Другие значения зарезервированы.	1
15:8	hp_mode	Данный байт передается в подключенную Flash память, в контрольные регистры, и в зависимости от модели активирует опции производительнсти в режимах Dual I/O и Quad I/O.	0
7:0	cmd	Тип команды на чтение: 8'h03 - Read; 8'h0B - Fast Read; 8'h3B - Fast Dual Out Read; 8'h6B - Fast Quad Out Read; 8'hBB - Fast Dual I/O Read; 8'hEB - Fast Quad I/O Read; Другие значения зарезервированы	3

9.7.8 qspi1_padcfg

Общий регистр настройки режима работы КП QSPI1.

Формат регистра qspi1_padcfg приведен в следующей таблице:

Таблица 9.18. Поля регистра qspi1_padcfg

Номер бита	Поле	Описание	Сброс
31:4	-	Резерв	0
3	od	Настройка OD КП	0

Номер бита	Поле	Описание	Сброс
2	cle	Настройка CLE КП	1
1	v18	Настройка V18 КП	0
0	en	Настройка EN КП	0

9.7.9 qspi1_ss_padcfg

Дополнительный регистр настройки КП QSPI1_SS (0, 1, 2, 3).

Формат регистра qspi1_ss_padcfg приведен в следующей таблице:

Таблица 9.19. Поля регистра qspi1_ss_padcfg

Номер бита	Поле	Описание	Сброс
31:16	-	Резерв	0
15	st	Настройка ST КП Влияет только на qspi1_ss0 при работе в режиме «вход»	0
14:11	-	Резерв	0
10:5	ctl	Настройка CTL КП	6'b000011
4:3	sl	Настройка SL КП	2'b11
2	pd	Настройка PD КП	0
1	pu	Настройка PU КП	0
0	sus	Настройка SUS КП	0

9.7.10 qspi1_siso_padcfg

Дополнительный регистр настройки КП QSPI1_SISO (0, 1, 2, 3).

Формат регистра qspi1_siso_padcfg приведен в следующей таблице:

Таблица 9.20. Поля регистра qspi1_siso_padcfg

Номер бита	Поле	Описание	Сброс
31:16	-	Резерв	0
15	st	Настройка ST КП	0
14:11	-	Резерв	0
10:5	ctl	Настройка CTL КП	6'b000011
4:3	sl	Настройка SL КП	2'b11
2	pd	Настройка PD КП	0
1	pu	Настройка PU КП	0
0	sus	Настройка SUS КП	0

9.7.11 qspi1_sclk_padcfg

Дополнительный регистр настройки КП QSPI1_SCLK.

Формат регистра qspi1_sclk_padcfg приведен в следующей таблице:

Таблица 9.21. Поля регистра qspi1_sclk_padcfg

Номер бита	Поле	Описание	Сброс
31:16	-	Резерв	0
15	st	Настройка ST КП	0
14:11	-	Резерв	0
10:5	ctl	Настройка CTL КП	6'b000011

Номер бита	Поле	Описание	Сброс
4:3	sl	Настройка SL КП	2'b11
2	pd	Настройка PD КП	0
1	pu	Настройка PU КП	0
0	sus	Настройка SUS КП	0

9.7.12sdmmc0_padcfg

Регистр настройки режима работы КП SDMMC0 кроме КП SDMMC0_18EN и SDMMC0_PWR.

Формат регистра sdmmc0_padcfg приведен в следующей таблице:

Таблица 9.22. Поля регистра sdmmc0_padcfg

Номер бита	Поле	Описание	Сброс
31:24	-	Резерв	0
23:4	v18delay	Задержка переключения настройки КП в режим V18. Значение по сбросу соответствует длительности в 1мс и частоте AXI sdmmc 100 МГц.	100000
3	use_sw_v18	Использовать бит v18 из данного регистра для настройки КП SDMMC	0
2	cle	Настройка CLE для всхем КП.	0
1	v18	Настройка V18 всех КП.	0
0	en	Настройка EN КП.	0

9.7.13sdmmc0_clk_padcfg

Дополнительный регистр настройки КП SDMMC0_CLK.

Формат регистра sdmmc0_clk_padcfg приведен в следующей таблице:

Таблица 9.23. Поля регистра sdmmc0_clk_padcfg

Номер бита	Поле	Описание	Сброс
31:18	-	Резерв	0
17	use_sw_ctl	Выбор источника управления входом CTL. 0 - см. таблицу в Разделе «Настройка типа КП» для SDMMC 1 - используется поле ctl данного регистра	0
16	od	Настройка OD КП	0
15:11	-	Резерв	0
10:5	ctl	Настройка CTL КП	6'b000011
4:3	sl	Настройка SL КП	2'b11
2	pd	Настройка PD КП	0
1	pu	Настройка PU КП	0
0	sus	Настройка SUS КП	0

9.7.14sdmmc0_cmd_padcfg

Дополнительный регистр настройки КП SDMMC0_CMD.

Формат регистра sdmmc0_cmd_padcfg приведен в следующей таблице:

Таблица 9.24. Поля регистра sdmmc0_cmd_padcfg

Номер бита	Поле	Описание	Сброс
31:18	-	Резерв	0
17	use_sw_ctl	Выбор источника управления входом CTL. 0 - см. таблицу в Разделе «Настройка типа КП» для SDMMC 1 - используется поле ctl данного регистра	0
16	od	Настройка OD КП	0
15	st	Настройка ST КП	0
14:11	-	Резерв	0
10:5	ctl	Настройка CTL КП	6'b000011
4:3	sl	Настройка SL КП	2'b11
2	pd	Настройка PD КП	0
1	pu	Настройка PU КП	0
0	sus	Настройка SUS КП	0

9.7.15 sdmmc0_dat_padcfg

Дополнительный регистр настройки КП SDMMC0_DAT (0 – 7).

Формат регистра sdmmc0_dat_padcfg приведен в следующей таблице:

Таблица 9.25. Поля регистра sdmmc0_dat_padcfg

Номер бита	Поле	Описание	Сброс
31:18	-	Резерв	0
17	use_sw_ctl	Выбор источника управления входом CTL. 0 - см. таблицу в Разделе «Настройка типа КП» для SDMMC 1 - используется поле ctl данного регистра	0
16	od	Настройка OD КП	0
15	st	Настройка ST КП	0
14:11	-	Резерв	0
10:5	ctl	Настройка CTL КП	6'b000011
4:3	sl	Настройка SL КП	2'b11
2	pd	Настройка PD КП	0
1	pu	Настройка PU КП	0
0	sus	Настройка SUS КП	0

9.7.16 sdmmc0_corecfg_0

Регистр 0 настроек SDMMC0. Определяет значение регистра SD Capabilities (044h).

Формат регистра sdmmc0_corecfg_0 приведен в следующей таблице:

Таблица 9.26. Поля регистра sdmmc0_corecfg_0

Номер бита	Поле	Описание	Сброс
31:26	-	Резерв	0
25	spiblkmode	См. описание регистра 'SD Capabilities'	0
24	spisupport	См. описание регистра 'SD Capabilities'	1
23:16	clockmultiplier	См. описание регистра 'SD Capabilities'	0
15:14	retuningmodes	См. описание регистра 'SD Capabilities'	0
13	tuningforsdr50	См. описание регистра 'SD Capabilities'	0
12	-	Резерв	0
11:8	retuningtimercnt	См. описание регистра 'SD Capabilities'	0

Номер бита	Поле	Описание	Сброс
7	-	Резерв	0
6	ddriversupport	См. описание регистра 'SD Capabilities'	1
5	cdriversupport	См. описание регистра 'SD Capabilities'	1
4	adriversupport	См. описание регистра 'SD Capabilities'	1
3	-	Резерв	0
2	ddr50support	См. описание регистра 'SD Capabilities'	1
1	sdr104support	См. описание регистра 'SD Capabilities'	1
0	sdr50support	См. описание регистра 'SD Capabilities'	1

9.7.17 sdmmc0_corecfg_1

Регистр 1 настроек SDMMC0. Определяет значение регистра SD Capabilities (040h).

Формат регистра sdmmc0_corecfg_1 приведен в следующей таблице:

Таблица 9.27. Поля регистра sdmmc0_corecfg_1

Номер бита	Поле	Описание	Сброс
31:30	slottype	См. описание регистра 'SD Capabilities'	0
29	asynchintr	См. описание регистра 'SD Capabilities'	1
28	addr_64bit	См. описание регистра 'SD Capabilities'	1
27	-	Резерв	0
26	volt_1p8	См. описание регистра 'SD Capabilities'	1
25	volt_3p0	См. описание регистра 'SD Capabilities'	0
24	volt_3p3	См. описание регистра 'SD Capabilities'	1
23	suspres	См. описание регистра 'SD Capabilities'	1
22	sdma	См. описание регистра 'SD Capabilities'	1
21	highspeed	См. описание регистра 'SD Capabilities'	1
20	-	Резерв	0
19	adma2	См. описание регистра 'SD Capabilities'	1
18	sdif_8bit	См. описание регистра 'SD Capabilities'	1
17:16	maxblklength	См. описание регистра 'SD Capabilities'	2'b10
15:8	baseclkfreq	См. описание регистра 'SD Capabilities'	25
7	timeoutclkunit	См. описание регистра 'SD Capabilities'	1
6	-	Резерв	0
5:0	timeoutclkfreq	См. описание регистра 'SD Capabilities'	0

9.7.18 sdmmc0_corecfg_2

Регистр 2 настроек SDMMC0. Определяет значение регистра SD Preset Value (060h).

Формат регистра sdmmc0_corecfg_2 приведен в следующей таблице:

Таблица 9.28. Поля регистра sdmmc0_corecfg_2

Номер бита	Поле	Описание	Сброс
31:29	-	Резерв	0
28:16	dspdpresetval	См. регистр 'SD Preset Value'	0
15:13	-	Резерв	0
12:0	initpresetval	См. регистр 'SD Preset Value'	0

9.7.19 sdmmc0_corecfg_3

Регистр 3 настроек SDMMC0. Определяет значение регистра SD Preset Value (064h).

Формат регистра sdmmc0_corecfg_3 приведен в следующей таблице:

Таблица 9.29. Поля регистра sdmmc0_corecfg_3

Номер бита	Поле	Описание	Сброс
31:29	-	Резерв	0
28:16	sdr12presetval	См. регистр 'SD Preset Value'	0
15:13	-	Резерв	0
12:0	hspdpresetval	См. регистр 'SD Preset Value'	0

9.7.20 sdmmc0_corecfg_4

Регистр 4 настроек SDMMC0. Определяет значение регистра SD Preset Value (068h).

Формат регистра sdmmc0_corecfg_4 приведен в следующей таблице:

Таблица 9.30. Поля регистра sdmmc0_corecfg_4

Номер бита	Поле	Описание	Сброс
31:29	-	Резерв	0
28:16	sdr50presetval	См. регистр 'SD Preset Value'	0
15:13	-	Резерв	0
12:0	sdr25presetval	См. регистр 'SD Preset Value'	0

9.7.21 sdmmc0_corecfg_5

Регистр 5 настроек SDMMC0. Определяет значение регистра SD Preset Value (06Ch).

Формат регистра sdmmc0_corecfg_5 приведен в следующей таблице:

Таблица 9.31. Поля регистра sdmmc0_corecfg_5

Номер бита	Поле	Описание	Сброс
31:29	-	Резерв	0
28:16	ddr50presetval	См. регистр 'SD Preset Value'	0
15:13	-	Резерв	0
12:0	sdr104presetval	См. регистр 'SD Preset Value'	0

9.7.22 sdmmc0_corecfg_6

Регистр 6 настроек SDMMC0. Определяет значение регистра SD Maximum Current Capabilities (048h).

Формат регистра sdmmc0_corecfg_6 приведен в следующей таблице:

Таблица 9.32. Поля регистра sdmmc0_corecfg_6

Номер бита	Поле	Описание	Сброс
31:24	-	Резерв	0
23:16	maxcurrent1p8v	См. регистр "SD Maximum Current Capabilities'	0
15:8	maxcurrent3p0v	См. регистр "SD Maximum Current Capabilities'	0
7:0	maxcurrent3p3v	См. регистр "SD Maximum Current Capabilities'	0

9.7.23sdmmc0_corecfg_7

Регистр 7 настроек SDMMC0. Управляет дополнительными входами SD.

Формат регистра sdmmc0_corecfg_7 приведен в следующей таблице:

Таблица 9.33. Поля регистра sdmmc0_corecfg_7

Номер бита	Поле	Описание	Сброс
31:20	-	Резерв	0
19	asyncwkupena	Управляет соответствующим входом контроллера	1
18:13	tuningcount	Управляет соответствующим входом контроллера	0x20
12	otapdlyena	Управляет соответствующим входом контроллера	0
11:8	otapdlysel	Управляет соответствующим входом контроллера	0
7	-	Резерв	0
6	itapchgwin	Управляет соответствующим входом контроллера	0
5	itapdlyena	Управляет соответствующим входом контроллера	0
4:0	itapdlysel	Управляет соответствующим входом контроллера	0

9.7.24sdmmc0_sdhc_dbg0

Регистр 0 отладки SDMMC0.

Формат регистра sdmmc0_sdhc_dbg0 приведен в следующей таблице:

Таблица 9.34. Поля регистра sdmmc0_sdhc_dbg0

Номер бита	Поле	Описание	Сброс
31:16	txddebugbus	См. соответствующий сигнал контроллера SDMMC	0
15:0	tundebugbus	См. соответствующий сигнал контроллера SDMMC	0

9.7.25sdmmc0_sdhc_dbg1

Регистр 1 отладки SDMMC0.

Формат регистра sdmmc0_sdhc_dbg1 приведен в следующей таблице:

Таблица 9.35. Поля регистра sdmmc0_sdhc_dbg1

Номер бита	Поле	Описание	Сброс
31:16	rxddebugbus1	См. соответствующий сигнал контроллера SDMMC	0
15:0	rxddebugbus0	См. соответствующий сигнал контроллера SDMMC	0

9.7.26 sdmmc0_sdhc_dbg2

Регистр 2 отладки SDMMC0.

Формат регистра sdmmc0_sdhc_dbg2 приведен в следующей таблице:

Таблица 9.36. Поля регистра sdmmc0_sdhc_dbg2

Номер бита	Поле	Описание	Сброс
31:16	dmadebugbus	См. соответствующий сигнал контроллера SDMMC	0
15:0	cmddebugbus	См. соответствующий сигнал контроллера SDMMC	0

9.7.27 sdmmc1_padcfg

Регистр настройки режима работы КП SDMMC1 кроме КП SDMMC1_18EN и SDMMC1_PWR.

Формат регистра sdmmc1_padcfg приведен в следующей таблице:

Таблица 9.37. Поля регистра sdmmc1_padcfg

Номер бита	Поле	Описание	Сброс
31:24	-	Резерв	0
23:4	v18delay	Задержка переключения настройки КП в режим V18. Значение по сбросу соответствует длительности в 1мс и частоте AXI sdmmc 100 МГц.	100000
3	use_sw_v18	Использовать бит v18 из данного регистра для настройки КП SDMMC	0
2	cle	Настройка CLE КП.	0
1	v18	Настройка V18	0
0	en	Настройка EN КП.	0

9.7.28 sdmmc1_clk_padcfg

Дополнительный регистр настройки КП SDMMC1_CLK.

Формат регистра sdmmc1_clk_padcfg приведен в следующей таблице:

Таблица 9.38. Поля регистра sdmmc1_clk_padcfg

Номер бита	Поле	Описание	Сброс
31:18	-	Резерв	0
17	use_sw_ctl	Выбор источника управления входом CTL. 0 - см. таблицу в Разделе «Настройка типа КП» для SDMMC 1 - используется поле ctl данного регистра	0
16	od	Настройка OD КП	0
15:11	-	Резерв	0
10:5	ctl	Настройка CTL КП	6'b000011
4:3	sl	Настройка SL КП	2'b11
2	pd	Настройка PD КП	0
1	pu	Настройка PU КП	0
0	sus	Настройка SUS КП	0

9.7.29sdmmc1_cmd_padcfg

Дополнительный регистр настройки КП SDMMC1_CMD.

Формат регистра sdmmc1_cmd_padcfg приведен в следующей таблице:

Таблица 9.39. Поля регистра sdmmc1_cmd_padcfg

Номер бита	Поле	Описание	Сброс
31:18	-	Резерв	0
17	use_sw_ctl	Выбор источника управления входом CTL. 0 - см. таблицу в Разделе «Настройка типа КП» для SDMMC 1 - используется поле ctl данного регистра	0
16	od	Настройка OD КП	0
15	st	Настройка ST КП	0
14:11	-	Резерв	0
10:5	ctl	Настройка CTL КП	6'b000011
4:3	sl	Настройка SL КП	2'b11
2	pd	Настройка PD КП	0
1	pu	Настройка PU КП	0
0	sus	Настройка SUS КП	0

9.7.30sdmmc1_dat_padcfg

Дополнительный регистр настройки КП SDMMC1_DAT (0 - 7).

Формат регистра sdmmc1_dat_padcfg приведен в следующей таблице:

Таблица 9.40. Поля регистра sdmmc1_dat_padcfg

Номер бита	Поле	Описание	Сброс
31:18	-	Резерв	0
17	use_sw_ctl	Выбор источника управления входом CTL. 0 - см. таблицу в Разделе «Настройка типа КП» для SDMMC 1 - используется поле ctl данного регистра	0
16	od	Настройка OD КП	0
15	st	Настройка ST КП	0
14:11	-	Резерв	0
10:5	ctl	Настройка CTL КП	6'b000011
4:3	sl	Настройка SL КП	2'b11
2	pd	Настройка PD КП	0
1	pu	Настройка PU КП	0
0	sus	Настройка SUS КП	0

9.7.31sdmmc1_corecfg_0

Регистр 0 настроек SDMMC1. Определяет значение регистра SD Capabilities (044h).

Формат регистра sdmmc1_corecfg_0 приведен в следующей таблице:

Таблица 9.41. Поля регистра sdmmc1_corecfg_0

Номер бита	Поле	Описание	Сброс
31:26	-	Резерв	0

Номер бита	Поле	Описание	Сброс
25	spiblkmode	См. описание регистра 'SD Capabilities'	0
24	spisupport	См. описание регистра 'SD Capabilities'	1
23:16	clockmultiplier	См. описание регистра 'SD Capabilities'	0
15:14	retuningmodes	См. описание регистра 'SD Capabilities'	0
13	tuningforsdr50	См. описание регистра 'SD Capabilities'	0
12	-	Резерв	0
11:8	retuningtimercnt	См. описание регистра 'SD Capabilities'	0
7	-	Резерв	0
6	ddriversupport	См. описание регистра 'SD Capabilities'	1
5	cdriversupport	См. описание регистра 'SD Capabilities'	1
4	adriversupport	См. описание регистра 'SD Capabilities'	1
3	-	Резерв	0
2	ddr50support	См. описание регистра 'SD Capabilities'	1
1	sdr104support	См. описание регистра 'SD Capabilities'	1
0	sdr50support	См. описание регистра 'SD Capabilities'	1

9.7.32sdmmc1_corecfg_1

Регистр 1 настроек SDMMC1. Определяет значение регистра SD Capabilities (040h).

Формат регистра sdmmc1_corecfg_1 приведен в следующей таблице:

Таблица 9.42. Поля регистра sdmmc1_corecfg_1

Номер бита	Поле	Описание	Сброс
31:30	slottype	См. описание регистра 'SD Capabilities'	0
29	asynchintr	См. описание регистра 'SD Capabilities'	1
28	addr_64bit	См. описание регистра 'SD Capabilities'	1
27	-	Резерв	0
26	volt_1p8	См. описание регистра 'SD Capabilities'	1
25	volt_3p0	См. описание регистра 'SD Capabilities'	0
24	volt_3p3	См. описание регистра 'SD Capabilities'	1
23	suspres	См. описание регистра 'SD Capabilities'	1
22	sdma	См. описание регистра 'SD Capabilities'	1
21	highspeed	См. описание регистра 'SD Capabilities'	1
20	-	Резерв	0
19	adma2	См. описание регистра 'SD Capabilities'	1
18	sdif_8bit	См. описание регистра 'SD Capabilities'	1
17:16	maxblklength	См. описание регистра 'SD Capabilities'	2'b10
15:8	baseclkfreq	См. описание регистра 'SD Capabilities'	25
7	timeoutclkunit	См. описание регистра 'SD Capabilities'	1
6	-	Резерв	0
5:0	timeoutclkfreq	См. описание регистра 'SD Capabilities'	0

9.7.33sdmmc1_corecfg_2

Регистр 2 настроек SDMMC1. Определяет значение регистра SD Preset Value (060h).

Формат регистра sdmmc1_corecfg_2 приведен в следующей таблице:

Таблица 9.43. Поля регистра sdmmc1_corecfg_2

Номер бита	Поле	Описание	Сброс
31:29	-	Резерв	0
28:16	dspdpresetval	См. регистр 'SD Preset Value'	0

Номер бита	Поле	Описание	Сброс
15:13	-	Резерв	0
12:0	initpresetval	См. регистр 'SD Preset Value'	0

9.7.34 sdmmc1_corecfg_3

Регистр 3 настроек SDMMC1. Определяет значение регистра SD Preset Value (064h).

Формат регистра sdmmc1_corecfg_3 приведен в следующей таблице:

Таблица 9.44. Поля регистра sdmmc1_corecfg_3

Номер бита	Поле	Описание	Сброс
31:29	-	Резерв	0
28:16	sdr12presetval	См. регистр 'SD Preset Value'	0
15:13	-	Резерв	0
12:0	hspdpresetval	См. регистр 'SD Preset Value'	0

9.7.35 sdmmc1_corecfg_4

Регистр 4 настроек SDMMC1. Определяет значение регистра SD Preset Value (068h).

Формат регистра sdmmc1_corecfg_4 приведен в следующей таблице:

Таблица 9.45. Поля регистра sdmmc1_corecfg_4

Номер бита	Поле	Описание	Сброс
31:29	-	Резерв	0
28:16	sdr50presetval	См. регистр 'SD Preset Value'	0
15:13	-	Резерв	0
12:0	sdr25presetval	См. регистр 'SD Preset Value'	0

9.7.36 sdmmc1_corecfg_5

Регистр 5 настроек SDMMC1. Определяет значение регистра SD Preset Value (06Ch).

Формат регистра sdmmc1_corecfg_5 приведен в следующей таблице:

Таблица 9.46. Поля регистра sdmmc1_corecfg_5

Номер бита	Поле	Описание	Сброс
31:29	-	Резерв	0
28:16	ddr50presetval	См. регистр 'SD Preset Value'	0
15:13	-	Резерв	0
12:0	sdr104presetval	См. регистр 'SD Preset Value'	0

9.7.37 sdmmc1_corecfg_6

Регистр 6 настроек SDMMC1. Определяет значение регистра SD Maximum Current Capabilities (048h).

Формат регистра sdmmc1_corecfg_6 приведен в следующей таблице:

Таблица 9.47. Поля регистра sdmmc1_corecfg_6

Номер бита	Поле	Описание	Сброс
31:24	-	Резерв	0
23:16	maxcurrent1p8v	См. регистр "SD Maximum Current Capabilities'	0
15:8	maxcurrent3p0v	См. регистр "SD Maximum Current Capabilities'	0
7:0	maxcurrent3p3v	См. регистр "SD Maximum Current Capabilities'	0

9.7.38 sdmmc1_corecfg_7

Регистр 7 настроек SDMMC1. Управляет дополнительными входами SD.

Формат регистра sdmmc1_corecfg_7 приведен в следующей таблице:

Таблица 9.48. Поля регистра sdmmc1_corecfg_7

Номер бита	Поле	Описание	Сброс
31:20	-	Резерв	0
19	asyncwkupena	Управляет соответствующим входом контроллера	0
18:13	tuningcount	Управляет соответствующим входом контроллера	0x20
12	otapdlyena	Управляет соответствующим входом контроллера	0
11:8	otapdlysel	Управляет соответствующим входом контроллера	0
7	-	Резерв	0
6	itapchgwin	Управляет соответствующим входом контроллера	0
5	itapdlyena	Управляет соответствующим входом контроллера	0
4:0	itapdlysel	Управляет соответствующим входом контроллера	0

9.7.39 sdmmc1_sdhc_dbg0

Регистр 0 отладки SDMMC1.

Формат регистра sdmmc1_sdhc_dbg0 приведен в следующей таблице:

Таблица 9.49. Поля регистра sdmmc1_sdhc_dbg0

Номер бита	Поле	Описание	Сброс
31:16	txddebugbus	См. соответствующий сигнал контроллера SDMMC	0
15:0	tundebugbus	См. соответствующий сигнал контроллера SDMMC	0

9.7.40 sdmmc1_sdhc_dbg1

Регистр 1 отладки SDMMC1.

Формат регистра sdmmc1_sdhc_dbg1 приведен в следующей таблице:

Таблица 9.50. Поля регистра sdmmc1_sdhc_dbg1

Номер бита	Поле	Описание	Сброс
31:16	rxddebugbus1	См. соответствующий сигнал контроллера SDMMC	0
15:0	rxddebugbus0	См. соответствующий сигнал контроллера SDMMC	0

9.7.41 sdmmc1_sdhc_dbg2

Регистр 2 отладки SDMMC1.

Формат регистра sdmmc1_sdhc_dbg2 приведен в следующей таблице:

Таблица 9.51. Поля регистра sdmmc1_sdhc_dbg2

Номер бита	Поле	Описание	Сброс
31:16	dmadebugbus	См. соответствующий сигнал контроллера SDMMC	0
15:0	cmddebugbus	См. соответствующий сигнал контроллера SDMMC	0

9.7.42 pdma2_dbg0

Регистр pdma2 dbg 0.

Формат регистра pdma2_dbg0 приведен в следующей таблице:

Таблица 9.52. Поля регистра pdma2_dbg0

Номер бита	Поле	Описание	Сброс
31:29	-	Резерв	0
28	dma_ctl_en	Состояние соответствующего внутреннего сигнала PDMA2	0
27	granted_m1	Состояние соответствующего внутреннего сигнала PDMA2	0
26	granted_m2	Состояние соответствующего внутреннего сигнала PDMA2	0
25	tfr_req_m1	Состояние соответствующего внутреннего сигнала PDMA2	0
24	tfr_req_m2	Состояние соответствующего внутреннего сигнала PDMA2	0
23:20	grant_index_m1	Состояние соответствующего внутреннего сигнала PDMA2	0
19:16	grant_index_m2	Состояние соответствующего внутреннего сигнала PDMA2	0
15:8	dum_req_src_region	Состояние соответствующего внутреннего сигнала PDMA2	0
7:0	dum_req_dst_region	Состояние соответствующего внутреннего сигнала PDMA2	0

9.7.43 pdma2_dbg1

Регистр pdma2 dbg 1.

Формат регистра pdma2_dbg1 приведен в следующей таблице:

Таблица 9.53. Поля регистра pdma2_dbg1

Номер бита	Поле	Описание	Сброс
31:24	fifo_ready_src	Состояние соответствующего внутреннего сигнала PDMA2	0
23:16	fifo_ready_dst	Состояние соответствующего внутреннего сигнала PDMA2	0
15:8	fifo_half_full	Состояние соответствующего внутреннего сигнала PDMA2	0
7:0	fifo_empty	Состояние соответствующего внутреннего сигнала PDMA2	0

9.7.44pdma2_dbg2

Регистр pdma2 dbg 2.

Формат регистра pdma2_dbg2 приведен в следующей таблице:

Таблица 9.54. Поля регистра pdma2_dbg2

Номер бита	Поле	Описание	Сброс
31:16	req_mi1	Состояние соответствующего внутреннего сигнала PDMA2	0
15:0	req_mi2	Состояние соответствующего внутреннего сигнала PDMA2	0

9.7.45pdma2_dbg3

Регистр pdma2 dbg 3.

Формат регистра pdma2_dbg3 приведен в следующей таблице:

Таблица 9.55. Поля регистра pdma2_dbg3

Номер бита	Поле	Описание	Сброс
31:24	rd_rawtfr	Состояние соответствующего внутреннего сигнала PDMA2	0
23:16	rd_rawblock	Состояние соответствующего внутреннего сигнала PDMA2	0
15:8	rd_rawsrctran	Состояние соответствующего внутреннего сигнала PDMA2	0
7:0	rd_rawdsttran	Состояние соответствующего внутреннего сигнала PDMA2	0

9.7.46pdma2_dbg4

Регистр pdma2 dbg 4.

Формат регистра pdma2_dbg4 приведен в следующей таблице:

Таблица 9.56. Поля регистра pdma2_dbg4

Номер бита	Поле	Описание	Сброс
31:24	-	Резерв	0
23:16	rd_rawerr	Состояние соответствующего внутреннего сигнала PDMA2	0

Номер бита	Поле	Описание	Сброс
15:8	rd_int_en	Состояние соответствующего внутреннего сигнала PDMA2	0
7:0	rd_maskerr	Состояние соответствующего внутреннего сигнала PDMA2	0

9.7.47 pdma2_dbg5

Регистр pdma2 dbg 5.

Формат регистра pdma2_dbg5 приведен в следующей таблице:

Таблица 9.57. Поля регистра pdma2_dbg5

Номер бита	Поле	Описание	Сброс
31:24	rd_masktfr	Состояние соответствующего внутреннего сигнала PDMA2	0
23:16	rd_maskblock	Состояние соответствующего внутреннего сигнала PDMA2	0
15:8	rd_masksrctran	Состояние соответствующего внутреннего сигнала PDMA2	0
7:0	rd_maskdsttran	Состояние соответствующего внутреннего сигнала PDMA2	0

9.7.48 pdma2_dbg6

Регистр pdma2 dbg 6.

Формат регистра pdma2_dbg6 приведен в следующей таблице:

Таблица 9.58. Поля регистра pdma2_dbg6

Номер бита	Поле	Описание	Сброс
31:16	mask_lck_ch_m1	Состояние соответствующего внутреннего сигнала PDMA2	0
15:0	mask_lck_ch_m2	Состояние соответствующего внутреннего сигнала PDMA2	0

9.7.49 pdma2_dbg7

Регистр pdma2 dbg 7.

Формат регистра pdma2_dbg7 приведен в следующей таблице:

Таблица 9.59. Поля регистра pdma2_dbg7

Номер бита	Поле	Описание	Сброс
31	-	Резерв	0
30:21	length_m_i	Состояние соответствующего внутреннего сигнала PDMA2	0
20:16	statusint_dmacore	Состояние соответствующего внутреннего сигнала PDMA2	0
15:8	en_src_hs_sgl	Состояние соответствующего внутреннего сигнала PDMA2	0
7:0	en_dst_hs_sgl	Состояние соответствующего внутреннего сигнала PDMA2	0

9.7.50pdma2_dbg8

Регистр pdma2 dbg 8.

Формат регистра pdma2_dbg8 приведен в следующей таблице:

Таблица 9.60. Поля регистра pdma2_dbg8

Номер бита	Поле	Описание	Сброс
31:16	dma_data_req	Состояние соответствующего внутреннего сигнала PDMA2	0
15:8	ch_enable_reg	Состояние соответствующего внутреннего сигнала PDMA2	0
7:0	ch_enable	Состояние соответствующего внутреннего сигнала PDMA2	0

9.7.51pdma2_bar0

Базовый адрес 0 axi PDMA2.

Формат регистра pdma2_bar0 приведен в следующей таблице:

Таблица 9.61. Поля регистра pdma2_bar0

Номер бита	Поле	Описание	Сброс
31:8	-	Резерв	0
7:0	addr	Разряды 39-32 адреса AXI.	0

9.7.52pdma2_bar1

Базовый адрес 1 axi PDMA2.

Формат регистра pdma2_bar1 приведен в следующей таблице:

Таблица 9.62. Поля регистра pdma2_bar1

Номер бита	Поле	Описание	Сброс
31:8	-	Резерв	0
7:0	addr	Разряды 39-32 адреса AXI.	0

9.7.53pdma2_bar2

Базовый адрес 2 axi PDMA2.

Формат регистра pdma2_bar2 приведен в следующей таблице:

Таблица 9.63. Поля регистра pdma2_bar2

Номер бита	Поле	Описание	Сброс
31:8	-	Резерв	0
7:0	addr	Разряды 39-32 адреса AXI.	0

9.7.54 pdma2_hmx_ctr

Регистр управления сигналами hmx.

Формат регистра pdma2_hmx_ctr приведен в следующей таблице:

Таблица 9.64. Поля регистра pdma2_hmx_ctr

Номер бита	Поле	Описание	Сброс
31:24	-	Резерв	0
23:20	nopx	Состояние выходов nopx мостов hmx.	4'hF
19:16	xwerrclr	Управление входами xwerrclr мостов hmx.	0
15:12	en_xwslverr	Разрешение прерываний по xwslverr.	0
11:8	en_xwdecerr	Разрешение прерываний по xwdecerr.	0
7:4	xwslverr	Состояние выходов xwslverr мостов hmx.	0
3:0	xwdecerr	Состояние выходов xwdecerr мостов hmx.	0

9.7.55 usb0_cntr_misc

Управление контроллером USB0.

Формат регистра usb0_cntr_misc приведен в следующей таблице:

Таблица 9.65. Поля регистра usb0_cntr_misc

Номер бита	Поле	Описание	Сброс
31:5	-	Резерв	0
4	host_port_power_control_present	Управляет входом host_port_power_control_present контроллера	1
3:0	bus_filter_bypass	Управление входом bus_filter_bypass контроллера USB	'b1111

9.7.56 usb0_phy_ctr

Регистр управления phy USB0. Управляющий регистр USB PHY.

Формат регистра usb0_phy_ctr приведен в следующей таблице:

Таблица 9.66. Поля регистра usb0_phy_ctr

Номер бита	Поле	Описание	Сброс
31:30	-	Резерв	0
29:24	fsel	Управление входом fsel[5:0] (Frequency Select) USP PHY.	6'b000010
23:17	mpll_multiplier	Управление входом mpll_multiplier[6:0] (MPLL Frequency Multiplier Control) USB PHY	7'b1100100
16	-	Резерв	0
15	ref_clkdiv2	Управление входом ref_clkdiv2 (Input Reference Clock Divider Control) USB PHY.	0
14	ref_ssp_en	Управление входом ref_ssp_en (Reference Clock Enable for SS function) USB PHY.	0
13	ref_use_pad	Управление входом ref_use_pad (Select Reference Clock Connected to ref_pad_clk_{p,m}) USB PHY.	0
12	ssc_en	Управление входом ssc_en (Spread Spectrum Enable) USB PHY.	0

Номер бита	Поле	Описание	Сброс
11:9	ssc_range	Управление входом ssc_range[2:0] (Spread Spectrum Clock Range) USB PHY.	0
8:0	ssc_ref_clk_sel	Управление входом ssc_ref_clk_sel[8:0] (Spread Spectrum Reference Clock Shifting) USB PHY.	0

9.7.57 usb0_phy_params0

Параметры phy USB0 0.

Формат регистра usb0_phy_params0 приведен в следующей таблице:

Таблица 9.67. Поля регистра usb0_phy_params0

Номер бита	Поле	Описание	Сброс
31	-	Резерв	0
30:28	compdistune0	Управляет входом COMPDISTUNE0[2:0] (Disconnect Threshold Adjustment) USB PHY.	3'b100
27:25	otgtune0	Управляет входом OTGTUNE0[2:0] (VBUS Valid Threshold Adjustment) USB PHY.	3'b100
24:22	sqrxtune0	Управляет входом SQRXTUNE0[2:0] (Squelch Threshold Adjustment) USB PHY.	3'b011
21:18	txfslstune0	Управляет входом TXFSLSTUNE0[3:0] (FS/LS Source Impedance Adjustment) USB PHY.	4'b0011
17:16	txhsxvtune0	Transmitter High-Speed Crossover Adjustment. Управляет входом TXHSXVTUNE0[1:0] USB PHY.	2'b11
15:14	txpreempamptune0	HS Transmitter Pre-Emphasis Current Control. Управляет входом TXPREEMPAMPTUNE0[1:0] USB PHY.	2'b00
13	txpreemppulsetune0	HS Transmitter Pre-Emphasis Duration Control. Управляет входом TXPREEMPPULSETUNE USB PHY.	1'b0
12:11	txrestune0	USB Source Impedance Adjustment. Управляет входом TXRESTUNE0[1:0] USB PHY.	2'b01
10:9	txrisetune0	HS Transmitter Rise/Fall Time Adjustment. Управляет входом TXRISETUNE0[1:0] USB PHY.	2'b10
8:5	txvreftune0	HS DC Voltage Level Adjustment. Управление вхом TXVREFTUNE0[3:0] USB PHY.	4'b1000
4:3	vdatreftune0	Data Detect Voltage Adjustment. Управление входом VDATREFTUNE0[1:0] USB PHY.	2'b01
2:0	los_bias	Loss-of-Signal Detector Threshold Level Control. Управление входом los_bias[2:0] USB PHY.	3'b100

9.7.58 usb0_phy_params1

Параметы phy USB0 1.

Формат регистра usb0_phy_params1 приведен в следующей таблице:

Таблица 9.68. Поля регистра usb0_phy_params1

Номер бита	Поле	Описание	Сброс
31:29	tx_vboost_lvl	TX Voltage Boost Level. Управляет входом tx_vboost_lvl[2:0] USB PHY.	3'b100

Номер бита	Поле	Описание	Сброс
28:22	pcs_tx_swing_full	TX Amplitude (Full Swing Mode). Управляет входом pcs_tx_swing_full[6:0] USB PHY.	127
21:16	pcs_tx_deemph_6db	TX De-Emphasis at 6 dB. Управляет входом pcs_tx_deemph_6db[5:0] USB PHY.	28
15:10	pcs_tx_deemph_3p5db	TX De-Emphasis at 3.5 dB. Управляет входом pcs_tx_deemph_3p5db[5:0] USB PHY.	28
9:0	pcs_rx_los_mask_val	Configurable Loss-of-Signal Mask Width. Управляет входом pcs_rx_los_mask_val[9:0] USB PHY.	0

9.7.59 usb0_phy_cr

Управляющий регистр phy USB0.

Формат регистра usb0_phy_cr приведен в следующей таблице:

Таблица 9.69. Поля регистра usb0_phy_cr

Номер бита	Поле	Описание	Сброс
31:21	-	Резерв	0
20	ack	CR Acknowledgment. Отображает состояние выхода cr_ack USB PHY.	0
19	read	CR Read. Подключён ко входу cr_read USB PHY.	0
18	write	CR Write. Подключён ко входу cr_write USB PHY.	0
17	cap_addr	CR Capture Address. Подключён ко входу cr_cap_addr USB PHY.	0
16	cap_data	CR Capture Data. Подключён ко входу cr_cap_data USB PHY.	0
15:0	data_io	При записи устанавливает входы cr_data_in[15:0] USB PHY. При чтении возвращает состояние выходов cr_data_out[15:0] USB PHY.	0

9.7.60 usb0_phy_tst

Регистр тестовых сигналов phy USB0.

Формат регистра usb0_phy_tst приведен в следующей таблице:

Таблица 9.70. Поля регистра usb0_phy_tst

Номер бита	Поле	Описание	Сброс
31:8	-	Резерв	0
7	idpullup	Управляет входом IDPULLUPO USB PHY	0
6	test_powerdown_ssp	Управляет входом test_powerdown_ssp USB PHY	0
5	test_powerdown_hsp	Управляет входом test_powerdown_hsp USB PHY	0
4	atereset	Управляет входом ATERESET USB PHY	0
3	loopbackenb	Управляет входом LOOPBACKENB0 USB PHY	0
2	lane0_tx2rx_loopbk	Управляет входом lane0_tx2rx_loopbk USB PHY	0
1	lane0_ext_pclk_req	Управляет входом lane0_ext_pclk_req USB PHY	0
0	vatestenb	Управляет входом VATESTENB USB PHY	0

9.7.61 usb0_phy_tst_bypass

Тест bypass phy USB0.

Формат регистра usb0_phy_tst_bypass приведен в следующей таблице:

Таблица 9.71. Поля регистра usb0_phy_tst_bypass

Номер бита	Поле	Описание	Сброс
31:8	-	Резерв	0
7	hsxcvrextctl	Управляет входом HSXCVREXTCTL0 USB PHY	0
6	hsrxdat	Возвращает значение выхода HSRXDAT0 USB PHY	0
5	hssquelch	Возвращает значение выхода HSSQUELCH0 USB PHY	0
4	sel	Управляет входом BYPASSSEL0 USB PHY	0
3	dmen	Управляет входом BYPASSDMEN0 USB PHY	0
2	dpen	Управляет входом BYPASSDPEN0 USB PHY	0
1	dmdata	Управляет входом BYPASSDMDATA0 USB PHY	0
0	dpdata	Управляет входом BYPASSDPDATA0 USB PHY	0

9.7.62 usb0_phy_rtune

Подстройка rtune USB0.

Формат регистра usb0_phy_rtune приведен в следующей таблице:

Таблица 9.72. Поля регистра usb0_phy_rtune

Номер бита	Поле	Описание	Сброс
31:2	-	Резерв	0
1	ack	Возвращает значение выхода rtune_ack USB PHY	0
0	req	Управляет входом rtune_req USB PHY	0

9.7.63 usb0_dbg0

Регистр отладки USB 0.

Формат регистра usb0_dbg0 приведен в следующей таблице:

Таблица 9.73. Поля регистра usb0_dbg0

Номер бита	Поле	Описание	Сброс
31:0	logic_analyzer_trace_31_0	Разряды 31:0 выхода контроллера USB logic_analyzer_trace	0

9.7.64 usb0_dbg1

Регистр отладки USB 1.

Формат регистра usb0_dbg1 приведен в следующей таблице:

Таблица 9.74. Поля регистра usb0_dbg1

Номер бита	Поле	Описание	Сброс
31:0	logic_analyzer_trace_63_32	Разряды 63:32 выхода контроллера USB logic_analyzer_trace	0

9.7.65 usb0_dbg2

Регистр отладки USB 2.

Формат регистра usb0_dbg2 приведен в следующей таблице:

Таблица 9.75. Поля регистра usb0_dbg2

Номер бита	Поле	Описание	Сброс
31:0	debug_31_0	Разряды 31:0 выхода контроллера USB debug	0

9.7.66 usb0_dbg3

Регистр отладки USB 3.

Формат регистра usb0_dbg3 приведен в следующей таблице:

Таблица 9.76. Поля регистра usb0_dbg3

Номер бита	Поле	Описание	Сброс
31:0	debug_63_32	Разряды 63:32 выхода контроллера USB debug	0

9.7.67 usb0_dbg4

Регистр отладки USB 4.

Формат регистра usb0_dbg4 приведен в следующей таблице:

Таблица 9.77. Поля регистра usb0_dbg4

Номер бита	Поле	Описание	Сброс
31:20	-	Резерв	0
19:17	clk_gate_ctrl	Выход контроллера USB clk_gate_ctrl	0
16:5	host_current_belt	Выход контроллера USB host_current_belt	0
4:3	operational_mode	Выход контроллера USB operational_mode	0
2:0	debug_66_64	Разряды 66:64 выхода debug контроллера USB	0

9.7.68 usb1_cntr_misc

Управление контроллером USB1.

Формат регистра usb1_cntr_misc приведен в следующей таблице:

Таблица 9.78. Поля регистра usb1_cntr_misc

Номер бита	Поле	Описание	Сброс
31:5	-	Резерв	0
4	host_port_power_control_present	Управляет входом host_port_power_control_present контроллера	1

Номер бита	Поле	Описание	Сброс
3:0	bus_filter_bypass	Управление входом bus_filter_bypass контроллера USB	'b1111

9.7.69 usb1_phy_ctr

Регистр управления phy USB1. Управляющий регистр USB PHY.

Формат регистра usb1_phy_ctr приведен в следующей таблице:

Таблица 9.79. Поля регистра usb1_phy_ctr

Номер бита	Поле	Описание	Сброс
31:30	-	Резерв	0
29:24	fsel	Управление входом fsel[5:0] (Frequency Select) USP PHY.	6'b000010
23:17	mpll_multiplier	Управление входом mpll_multiplier[6:0] (MPLL Frequency Multiplier Control) USB PHY	7'b1100100
16	-	Резерв	0
15	ref_clkdiv2	Управление входом ref_clkdiv2 (Input Reference Clock Divider Control) USB PHY.	0
14	ref_ssp_en	Управление входом ref_ssp_en (Reference Clock Enable for SS function) USB PHY.	0
13	ref_use_pad	Управление входом ref_use_pad (Select Reference Clock Connected to ref_pad_clk_{p,m}) USB PHY.	0
12	ssc_en	Управление входом ssc_en (Spread Spectrum Enable) USB PHY.	0
11:9	ssc_range	Управление входом ssc_range[2:0] (Spread Spectrum Clock Range) USB PHY.	0
8:0	ssc_ref_clk_sel	Управление входом ssc_ref_clk_sel[8:0] (Spread Spectrum Reference Clock Shifting) USB PHY.	0

9.7.70 usb1_phy_params0

Параметры phy USB1 0.

Формат регистра usb1_phy_params0 приведен в следующей таблице:

Таблица 9.80. Поля регистра usb1_phy_params0

Номер бита	Поле	Описание	Сброс
31	-	Резерв	0
30:28	compdistune0	Управляет входом COMPDISTUNE0[2:0] (Disconnect Threshold Adjustment) USB PHY.	3'b100
27:25	otgtune0	Управляет входом OTGTUNE0[2:0] (VBUS Valid Threshold Adjustment) USB PHY.	3'b100
24:22	sqrxtune0	Управляет входом SQRXTUNE0[2:0] (Squelch Threshold Adjustment) USB PHY.	3'b011
21:18	txfslstune0	Управляет входом TXFSLSTUNE0[3:0] (FS/LS Source Impedance Adjustment) USB PHY.	4'b0011
17:16	txhsxvtune0	Transmitter High-Speed Crossover Adjustment. Управляет входом TXHSXVTUNE0[1:0] USB PHY.	2'b11
15:14	txpreempamptune0	HS Transmitter Pre-Emphasis Current Control. Управляет входом TXPREEMPAMPTUNE0[1:0] USB PHY.	2'b00

Номер бита	Поле	Описание	Сброс
13	txpreemppulsetune0	HS Transmitter Pre-Emphasis Duration Control. Управляет входом TXPREEMPPULSETUNE USB PHY.	1'b0
12:11	txrestune0	USB Source Impedance Adjustment. Управляет входом TXRESTUNE0[1:0] USB PHY.	2'b01
10:9	txrisetune0	HS Transmitter Rise/Fall Time Adjustment. Управляет входом TXRISETUNE0[1:0] USB PHY.	2'b10
8:5	txvreftune0	HS DC Voltage Level Adjustment. Управление вхом TXVREFTUNE0[3:0] USB PHY.	4'b1000
4:3	vdatreftune0	Data Detect Voltage Adjustment. Управление входом VDATREFTUNE0[1:0] USB PHY.	2'b01
2:0	los_bias	Loss-of-Signal Detector Threshold Level Control. Управление входом los_bias[2:0] USB PHY.	3'b100

9.7.71usb1_phy_params1

Параметры phy USB1 1.

Формат регистра usb1_phy_params1 приведен в следующей таблице:

Таблица 9.81. Поля регистра usb1_phy_params1

Номер бита	Поле	Описание	Сброс
31:29	tx_vboost_lvl	TX Voltage Boost Level. Управляет входом tx_vboost_lvl[2:0] USB PHY.	3'b100
28:22	pcs_tx_swing_full	TX Amplitude (Full Swing Mode). Управляет входом pcs_tx_swing_full[6:0] USB PHY.	127
21:16	pcs_tx_deemph_6db	TX De-Emphasis at 6 dB. Управляет входом pcs_tx_deemph_6db[5:0] USB PHY.	28
15:10	pcs_tx_deemph_3p5db	TX De-Emphasis at 3.5 dB. Управляет входом pcs_tx_deemph_3p5db[5:0] USB PHY.	28
9:0	pcs_rx_los_mask_val	Configurable Loss-of-Signal Mask Width. Управляет входом pcs_rx_los_mask_val[9:0] USB PHY.	0

9.7.72usb1_phy_cr

Управляющий регистр phy USB1.

Формат регистра usb1_phy_cr приведен в следующей таблице:

Таблица 9.82. Поля регистра usb1_phy_cr

Номер бита	Поле	Описание	Сброс
31:21	-	Резерв	0
20	ack	CR Acknowledgment. Отображает состояние выхода cr_ack USB PHY.	0
19	read	CR Read. Подключён ко входу cr_read USB PHY.	0
18	write	CR Write. Подключён ко входу cr_write USB PHY.	0
17	cap_addr	CR Capture Address. Подключён ко входу cr_cap_addr USB PHY.	0
16	cap_data	CR Capture Data. Подключён ко входу cr_cap_data USB PHY.	0

Номер бита	Поле	Описание	Сброс
15:0	data_io	При записи устанавливает входы cr_data_in[15:0] USB PHY. При чтении возвращает состояние выходов cr_data_out[15:0] USB PHY.	0

9.7.73usb1_phy_tst

Регистр тестовых сигналов phy USB1.

Формат регистра usb1_phy_tst приведен в следующей таблице:

Таблица 9.83. Поля регистра usb1_phy_tst

Номер бита	Поле	Описание	Сброс
31:8	-	Резерв	0
7	idpullup	Управляет входом IDPULLUPO USB PHY	0
6	test_powerdown_ssp	Управляет входом test_powerdown_ssp USB PHY	0
5	test_powerdown_hsp	Управляет входом test_powerdown_hsp USB PHY	0
4	atereset	Управляет входом ATERESET USB PHY	0
3	loopbackenb	Управляет входом LOOPBACKENB0 USB PHY	0
2	lane0_tx2rx_loopbk	Управляет входом lane0_tx2rx_loopbk USB PHY	0
1	lane0_ext_pclk_req	Управляет входом lane0_ext_pclk_req USB PHY	0
0	vatestenb	Управляет входом VATESTENB USB PHY	0

9.7.74usb1_phy_tst_bypass

Тест bypass phy USB1.

Формат регистра usb1_phy_tst_bypass приведен в следующей таблице:

Таблица 9.84. Поля регистра usb1_phy_tst_bypass

Номер бита	Поле	Описание	Сброс
31:8	-	Резерв	0
7	hsxcvrextctl	Управляет входом HSXCVREXTCTL0 USB PHY	0
6	hsrxdat	Возвращает значение выхода HSRXDAT0 USB PHY	0
5	hssquelch	Возвращает значение выхода HSSQUELCH0 USB PHY	0
4	sel	Управляет входом BYPASSSEL0 USB PHY	0
3	dmen	Управляет входом BYPASSDMEN0 USB PHY	0
2	dpen	Управляет входом BYPASSDPEN0 USB PHY	0
1	dmdata	Управляет входом BYPASSDMDATA0 USB PHY	0
0	dpdata	Управляет входом BYPASSDPDATA0 USB PHY	0

9.7.75usb1_phy_rtune

Подстройка rtune USB1.

Формат регистра usb1_phy_rtune приведен в следующей таблице:

Таблица 9.85. Поля регистра usb1_phy_rtune

Номер бита	Поле	Описание	Сброс
31:2	-	Резерв	0
1	ack	Возвращает значение выхода rtune_ack USB PHY	0
0	req	Управляет входом rtune_req USB PHY	0

9.7.76 usb1_dbg0

Регистр отладки USB 0.

Формат регистра usb1_dbg0 приведен в следующей таблице:

Таблица 9.86. Поля регистра usb1_dbg0

Номер бита	Поле	Описание	Сброс
31:0	logic_analyzer_trace_31_0	Разряды 31:0 выхода контроллера USB logic_analyzer_trace	0

9.7.77 usb1_dbg1

Регистр отладки USB 1.

Формат регистра usb1_dbg1 приведен в следующей таблице:

Таблица 9.87. Поля регистра usb1_dbg1

Номер бита	Поле	Описание	Сброс
31:0	logic_analyzer_trace_63_32	Разряды 63:32 выхода контроллера USB logic_analyzer_trace	0

9.7.78 usb1_dbg2

Регистр отладки USB 2.

Формат регистра usb1_dbg2 приведен в следующей таблице:

Таблица 9.88. Поля регистра usb1_dbg2

Номер бита	Поле	Описание	Сброс
31:0	debug_31_0	Разряды 31:0 выхода контроллера USB debug	0

9.7.79 usb1_dbg3

Регистр отладки USB 3.

Формат регистра usb1_dbg3 приведен в следующей таблице:

Таблица 9.89. Поля регистра usb1_dbg3

Номер бита	Поле	Описание	Сброс
31:0	debug_63_32	Разряды 63:32 выхода контроллера USB debug	0

9.7.80 usb1_dbg4

Регистр отладки USB 4.

Формат регистра usb1_dbg4 приведен в следующей таблице:

Таблица 9.90. Поля регистра usb1_dbg4

Номер бита	Поле	Описание	Сброс
31:20	-	Резерв	0
19:17	clk_gate_ctrl	Выход контроллера USB clk_gate_ctrl	0
16:5	host_current_belt	Выход контроллера USB host_current_belt	0
4:3	operational_mode	Выход контроллера USB operational_mode	0
2:0	debug_66_64	Разряды 66:64 выхода debug контроллера USB	0

9.7.81 emac_padcfg

Общий регистр настройки режима работы КП обоих контроллеров EMAC0 и EMAC1.

Формат регистра emac_padcfg приведен в следующей таблице:

Таблица 9.91. Поля регистра emac_padcfg

Номер бита	Поле	Описание	Сброс
31:1	-	Резерв	0
0	v18	Настройка V18 КП EMAC.	0

9.7.82 emac0_padcfg

Дополнительный регистр настройки КП EMAC0.

Формат регистра emac0_padcfg приведен в следующей таблице:

Таблица 9.92. Поля регистра emac0_padcfg

Номер бита	Поле	Описание	Сброс
31:4	-	Резерв	0
3	od	Настройка OD КП	0
2:1	-	Резерв	0
0	en	Настройка EN КП	0

9.7.83 emac0_tx_padcfg

Дополнительный регистр настройки КП EMAC0_RGMII_TXD*, EMAC0_RGMII_TXCTL.

Формат регистра emac0_tx_padcfg приведен в следующей таблице:

Таблица 9.93. Поля регистра emac0_tx_padcfg

Номер бита	Поле	Описание	Сброс
31:11	-	Резерв	0
10:5	ctl	Настройка CTL КП	6'b000011
4:3	sl	Настройка SL КП	2'b11
2	pd	Настройка PD КП	0
1	pu	Настройка PU КП	0

Номер бита	Поле	Описание	Сброс
0	sus	Настройка SUS КП	0

9.7.84 emac0_txc_padcfg

Дополнительный регистр настройки КП EMAC0_RGMII_TXC.

Формат регистра emac0_txc_padcfg приведен в следующей таблице:

Таблица 9.94. Поля регистра emac0_txc_padcfg

Номер бита	Поле	Описание	Сброс
31:11	-	Резерв	0
10:5	ctl	Настройка CTL КП	6'b000011
4:3	sl	Настройка SL КП	2'b11
2	pd	Настройка PD КП	0
1	pu	Настройка PU КП	0
0	sus	Настройка SUS КП	0

9.7.85 emac0_rx_padcfg

Дополнительный регистр настройки КП EMAC0_RGMII_RXD*, EMAC0_RGMII_RXCTL.

Формат регистра emac0_rx_padcfg приведен в следующей таблице:

Таблица 9.95. Поля регистра emac0_rx_padcfg

Номер бита	Поле	Описание	Сброс
31:16	-	Резерв	0
15	st	Настройка ST КП	0
14:3	-	Резерв	0
2	pd	Настройка PD КП	0
1	pu	Настройка PU КП	0
0	sus	Настройка SUS КП	0

9.7.86 emac0_rxc_padcfg

Дополнительный регистр настройки КП EMAC0_RGMII_RXC.

Формат регистра emac0_rxc_padcfg приведен в следующей таблице:

Таблица 9.96. Поля регистра emac0_rxc_padcfg

Номер бита	Поле	Описание	Сброс
31:16	-	Резерв	0
15	st	Настройка ST КП	0
14:3	-	Резерв	0
2	pd	Настройка PD КП	0
1	pu	Настройка PU КП	0
0	sus	Настройка SUS КП	0

9.7.87 emac0_md_padcfg

Дополнительный регистр настройки КП EMAC0_RGMII_MDIO.

Формат регистра emac0_md_padcfg приведен в следующей таблице:

Таблица 9.97. Поля регистра emac0_md_padcfg

Номер бита	Поле	Описание	Сброс
31:16	-	Резерв	0
15	st	Настройка ST КП	0
14:11	-	Резерв	0
10:5	ctl	Настройка CTL КП	6'b000011
4:3	sl	Настройка SL КП	2'b11
2	pd	Настройка PD КП	0
1	pu	Настройка PU КП	0
0	sus	Настройка SUS КП	0

9.7.88 emac0_mdc_padcfg

Дополнительный регистр настройки КП EMAC0_RGMII_MDC.

Формат регистра emac0_mdc_padcfg приведен в следующей таблице:

Таблица 9.98. Поля регистра emac0_mdc_padcfg

Номер бита	Поле	Описание	Сброс
31:11	-	Резерв	0
10:5	ctl	Настройка CTL КП	6'b000011
4:3	sl	Настройка SL КП	2'b11
2	pd	Настройка PD КП	0
1	pu	Настройка PU КП	0
0	sus	Настройка SUS КП	0

9.7.89 emac0_bar

Базовый адрес axi EMAC0.

Формат регистра emac0_bar приведен в следующей таблице:

Таблица 9.99. Поля регистра emac0_bar

Номер бита	Поле	Описание	Сброс
31:8	-	Резерв	0
7:0	addr	Разряды 39-32 адреса AXI.	0

9.7.90 emac1_padcfg

Дополнительный регистр настройки КП EMAC1.

Формат регистра emac1_padcfg приведен в следующей таблице:

Таблица 9.100. Поля регистра emac1_padcfg

Номер бита	Поле	Описание	Сброс
31:4	-	Резерв	0
3	od	Настройка OD КП	0
2:1	-	Резерв	0
0	en	Настройка EN КП	0

9.7.91 emac1_tx_padcfg

Дополнительный регистр настройки КП EMAC1_RGMII_TXD*, EMAC1_RGMII_TXCTL.

Формат регистра emac1_tx_padcfg приведен в следующей таблице:

Таблица 9.101. Поля регистра emac1_tx_padcfg

Номер бита	Поле	Описание	Сброс
31:11	-	Резерв	0
10:5	ctl	Настройка CTL КП	6'b000011
4:3	sl	Настройка SL КП	2'b11
2	pd	Настройка PD КП	0
1	pu	Настройка PU КП	0
0	sus	Настройка SUS КП	0

9.7.92 emac1_txc_padcfg

Дополнительный регистр настройки КП EMAC1_RGMII_TXC.

Формат регистра emac1_txc_padcfg приведен в следующей таблице:

Таблица 9.102. Поля регистра emac1_txc_padcfg

Номер бита	Поле	Описание	Сброс
31:11	-	Резерв	0
10:5	ctl	Настройка CTL КП	6'b000011
4:3	sl	Настройка SL КП	2'b11
2	pd	Настройка PD КП	0
1	pu	Настройка PU КП	0
0	sus	Настройка SUS КП	0

9.7.93 emac1_rx_padcfg

Дополнительный регистр настройки КП EMAC1_RGMII_RXD*, EMAC1_RGMII_RXCTL.

Формат регистра emac1_rx_padcfg приведен в следующей таблице:

Таблица 9.103. Поля регистра emac1_rx_padcfg

Номер бита	Поле	Описание	Сброс
31:16	-	Резерв	0
15	st	Настройка ST КП	0
14:3	-	Резерв	0
2	pd	Настройка PD КП	0
1	pu	Настройка PU КП	0
0	sus	Настройка SUS КП	0

9.7.94 emac1_rxc_padcfg

Дополнительный регистр настройки КП EMAC1_RGMII_RXC.

Формат регистра emac1_rxc_padcfg приведен в следующей таблице:

Таблица 9.104. Поля регистра emac1_rxc_padcfg

Номер бита	Поле	Описание	Сброс
31:16	-	Резерв	0
15	st	Настройка ST КП	0
14:3	-	Резерв	0
2	pd	Настройка PD КП	0
1	pu	Настройка PU КП	0
0	sus	Настройка SUS КП	0

9.7.95 emac1_md_padcfg

Дополнительный регистр настройки КП EMAC1_RGMII_MDIO.

Формат регистра emac1_md_padcfg приведен в следующей таблице:

Таблица 9.105. Поля регистра emac1_md_padcfg

Номер бита	Поле	Описание	Сброс
31:16	-	Резерв	0
15	st	Настройка ST КП	0
14:11	-	Резерв	0
10:5	ctl	Настройка CTL КП	6'b000011
4:3	sl	Настройка SL КП	2'b11
2	pd	Настройка PD КП	0
1	pu	Настройка PU КП	0
0	sus	Настройка SUS КП	0

9.7.96 emac1_mdc_padcfg

Дополнительный регистр настройки КП EMAC1_RGMII_MDC.

Формат регистра emac1_mdc_padcfg приведен в следующей таблице:

Таблица 9.106. Поля регистра emac1_mdc_padcfg

Номер бита	Поле	Описание	Сброс
31:11	-	Резерв	0
10:5	ctl	Настройка CTL КП	6'b000011
4:3	sl	Настройка SL КП	2'b11
2	pd	Настройка PD КП	0
1	pu	Настройка PU КП	0
0	sus	Настройка SUS КП	0

9.7.97 emac1_bar

Базовый адрес axi EMAC1.

Формат регистра emac1_bar приведен в следующей таблице:

Таблица 9.107. Поля регистра emac1_bar

Номер бита	Поле	Описание	Сброс
31:8	-	Резерв	0
7:0	addr	Разряды 39-32 адреса AXI.	0

9.7.98 nand_padcfg

Общий регистр настройки КП NAND.

Формат регистра nand_padcfg приведен в следующей таблице:

Таблица 9.108. Поля регистра nand_padcfg

Номер бита	Поле	Описание	Сброс
31:5	-	Резерв	0
4	off_io_share	Разрешение использования NFC_IO* блоком GNSS. 1 - Контроллер NAND управляет состоянием NFC_IO*. 0 - Контроллер NAND не влияет на состояние NFC_IO*(КП используются только как входы)	1
3	od	Настройка OD КП	0
2	cle	Настройка CLE КП	1
1	v18	Настройка V18 КП NAND.	0
0	en	Настройка EN КП	0

9.7.99 nand_ale_cle_padcfg

Дополнительный регистр настройки КП NFC_ALE, NFC_CLE.

Формат регистра nand_ale_cle_padcfg приведен в следующей таблице:

Таблица 9.109. Поля регистра nand_ale_cle_padcfg

Номер бита	Поле	Описание	Сброс
31:11	-	Резерв	0
10:5	ctl	Настройка CTL КП	6'b000011
4:3	sl	Настройка SL КП	2'b11
2	pd	Настройка PD КП	0
1	pu	Настройка PU КП	0
0	sus	Настройка SUS КП	0

9.7.100 nand_cen_padcfg

Дополнительный регистр настройки КП NFC_CEN0, NFC_CEN1.

Формат регистра nand_cen_padcfg приведен в следующей таблице:

Таблица 9.110. Поля регистра nand_cen_padcfg

Номер бита	Поле	Описание	Сброс
31:11	-	Резерв	0
10:5	ctl	Настройка CTL КП	6'b000011
4:3	sl	Настройка SL КП	2'b11
2	pd	Настройка PD КП	0
1	pu	Настройка PU КП	0
0	sus	Настройка SUS КП	0

9.7.101 nand_ren_padcfg

Дополнительный регистр настройки КП NFC_REN.

Формат регистра nand_ren_padcfg приведен в следующей таблице:

Таблица 9.111. Поля регистра nand_ren_padcfg

Номер бита	Поле	Описание	Сброс
31:11	-	Резерв	0
10:5	ctl	Настройка CTL КП	6'b000011
4:3	sl	Настройка SL КП	2'b11
2	pd	Настройка PD КП	0
1	pu	Настройка PU КП	0
0	sus	Настройка SUS КП	0

9.7.102 nand_wen_padcfg

Дополнительный регистр настройки КП NFC_WEN.

Формат регистра nand_wen_padcfg приведен в следующей таблице:

Таблица 9.112. Поля регистра nand_wen_padcfg

Номер бита	Поле	Описание	Сброс
31:11	-	Резерв	0
10:5	ctl	Настройка CTL КП	6'b000011
4:3	sl	Настройка SL КП	2'b11
2	pd	Настройка PD КП	0
1	pu	Настройка PU КП	0
0	sus	Настройка SUS КП	0

9.7.103 nand_io_padcfg

Дополнительный регистр настройки КП NFC_IO (0 – 15).

Формат регистра nand_io_padcfg приведен в следующей таблице:

Таблица 9.113. Поля регистра nand_io_padcfg

Номер бита	Поле	Описание	Сброс
31:16	-	Резерв	0
15	st	Настройка ST КП	0
14:11	-	Резерв	0
10:5	ctl	Настройка CTL КП	6'b000011
4:3	sl	Настройка SL КП	2'b11
2	pd	Настройка PD КП	0
1	pu	Настройка PU КП	0
0	sus	Настройка SUS КП	0

9.7.104 nand_dqs_padcfg

Дополнительный регистр настройки КП NFC_DQS.

Формат регистра nand_dqs_padcfg приведен в следующей таблице:

Таблица 9.114. Поля регистра nand_dqs_padcfg

Номер бита	Поле	Описание	Сброс
31:16	-	Резерв	0
15	st	Настройка ST КП	0
14:11	-	Резерв	0
10:5	ctl	Настройка CTL КП	6'b000011
4:3	sl	Настройка SL КП	2'b11
2	pd	Настройка PD КП	0
1	pu	Настройка PU КП	0
0	sus	Настройка SUS КП	0

9.7.105 nand_rbn_padcfg

Дополнительный регистр настройки КП NFC_RBN0, NFC_RBN1.

Формат регистра nand_rbn_padcfg приведен в следующей таблице:

Таблица 9.115. Поля регистра nand_rbn_padcfg

Номер бита	Поле	Описание	Сброс
31:16	-	Резерв	0
15	st	Настройка ST КП	0
14:11	-	Резерв	0
10:5	ctl	Настройка CTL КП	6'b000011
4:3	sl	Настройка SL КП	2'b11
2	pd	Настройка PD КП	0
1	pu	Настройка PU КП	0
0	sus	Настройка SUS КП	0

9.7.106 misc_padcfg

Регистр настройки КП MISC.

Формат регистра misc_padcfg приведен в следующей таблице:

Таблица 9.116. Поля регистра misc_padcfg

Номер бита	Поле	Описание	Сброс
31:16	-	Резерв	0
15	usb_ocn_filt_en	Включение фильтрации входного сигнала USB EN_OCn. 0 - на вход контроллера USB hub_port_overcurrent заводится '0'. 1 - на вход контроллера USB hub_port_overcurrent заводится отфильтрованная версия сигнала EN_OCn.	1
14	wp_pu	Настройка PU КП SDMMC*_WP.	1
13	wp_pd	Настройка PD КП SDMMC*_WP.	0
12	wp_sus	Настройка SUS КП SDMMC*_WP.	0
11	cdn_pu	Настройка PU КП SDMMC*_CDN.	1
10	cdn_pd	Настройка PD КП SDMMC*_CDN.	0
9	cdn_sus	Настройка SUS КП SDMMC*_CDN.	0
8	en	Настройка EN КП группы MISC.	0
7	v18	Настройка V18 всех КП группы MISC.	0
6	cle	Настройка CLE КП	1
5:0	ctl	Настройка CTL КП	6'b000011

9.7.107 dbg_ctr

Регистр отладки.

Формат регистра dbg_ctr приведен в следующей таблице:

Таблица 9.117. Поля регистра dbg_ctr

Номер бита	Поле	Описание	Сброс
31:3	-	Резерв	0
2	force_dbg_usb_en_to_1	Устанавливает значение dbg_usb_en в 1 независимо от входов hsperiph.	0
1	force_dbg_usb_en_to_0	Устанавливает значение dbg_usb_en в 0 независимо от входов hsperiph.	0
0	usb_attached	Разрешение работы USB контроллера Synopsys совместно с режимом отладки. Поле необходимо установить в 1 после того, как контроллера настроен и готов к работе.	0

9.7.108 dma_axcache

Регистр настройки сигналов AXCACHE DMA.

Формат регистра dma_axcache приведен в следующей таблице:

Таблица 9.118. Поля регистра dma_axcache

Номер бита	Поле	Описание	Сброс
31:28	dbg_sysbuf	Определяет значение AXCACHE для DMA DEBUG.SYS BUF	0
27:24	dbg_dma	Определяет значение AXCACHE для DMA DEBUG.DMA	0
23:20	pdma	Определяет значение AXCACHE для DMA PDMA2	0
19:16	nand	Определяет значение AXCACHE для DMA NAND	0
15:12	sdmmc1	Определяет значение AXCACHE для DMA SDMMC1	0
11:8	sdmmc0	Определяет значение AXCACHE для DMA SDMMC0	0
7:4	emac1	Определяет значение AXCACHE для DMA EMAC1	0
3:0	emac0	Определяет значение AXCACHE для DMA EMAC0	0

9.8 EMAC0, EMAC1

На следующем рисунке показаны особенности интеграция блока EMAC0:

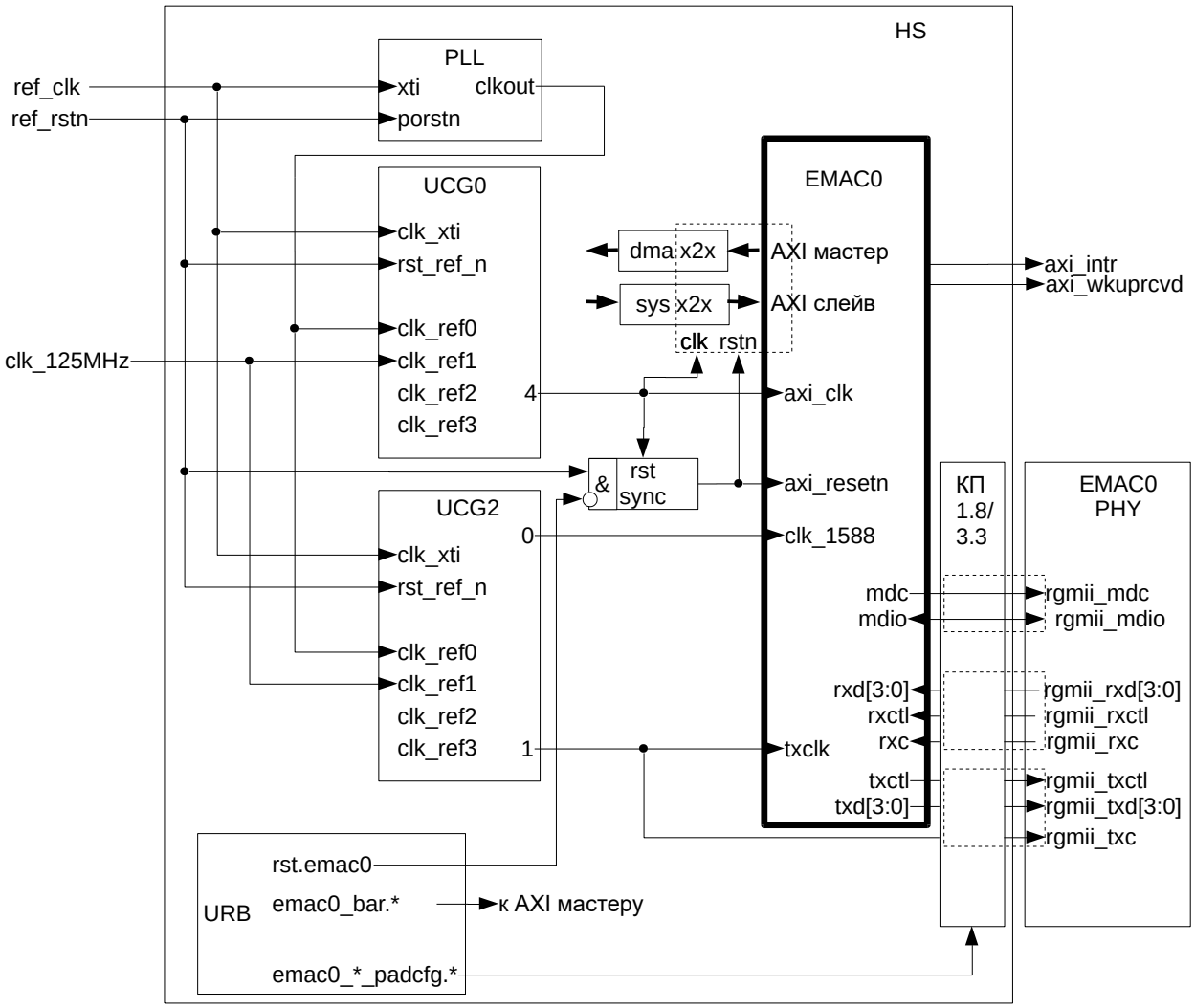


Рисунок 9.7 HSPERIPH. Особенности интеграции EMAC0

Замечание. EMAC0 PHY внешний для микросхемы компонент.

9.8.1 Состояние после сброса по ref_clk

После выхода из сброса по ref_clk блоки EMAC находятся в состоянии:

- Контактные площадки отключены (все выходы в 3-ем состоянии).
- Блок находится под сбросом, управляемым из регистра URB.rst.emac0.
- Частоты axi_clk, clk_1558 и txclk отключены.
- В качестве опорной частоты UCG2 выбран вход clk_ref1 – CLK125.

9.8.2 Последовательность настройки после выхода из сброса по ref_clk

После снятия сигнал сброса настройку блока следует производить в следующем порядке:

1. Настройка и включение частот axi_clk, txclk, clk_1588.
2. Снятие сигнала сброса URB.rst.emac* по истечению 64 тактов после включения частоты axi_clk.
3. Настройка КП на необходимый режим 1.8В или 3.3В.
4. Включение КП (вывод выходов из 3-го состояния).
5. Дальнейшая настройка EMAC* через регистры EMAC*.

9.8.3 Сброс emac через регистры URB.rst.emac*

Сигнал сброса axi_resetn должен удерживаться не менее 64 тактов частоты axi_clk.

9.8.4 Базовые адресные регистры

Мастер DMA EMAC работает в 32-х разрядном адресном пространстве. Для работы с 40-ка разрядным адресом, предусмотрены регистры URB.emac*_bar. Разряды 39:32 адреса при обращение к коммутатору DMA напрямую берутся из регистров URB.emac*_bar

Внимание. Нельзя производить запись в регистры URB.emac*_bar при активном DMA EMAC*. Если требуется изменить настройки URB.emac*_bar, необходимо:

1. Дождаться, когда закончится активность axi мастера EMAC*.
2. Обновить содержимое регистров URB.emac*_bar.
3. Выдержать паузу достаточную для обновления регистров.
4. Запустить мастер DMA в работу.

9.9 USB0, USB1

На следующем рисунке показаны особенности интеграция блока USB0:

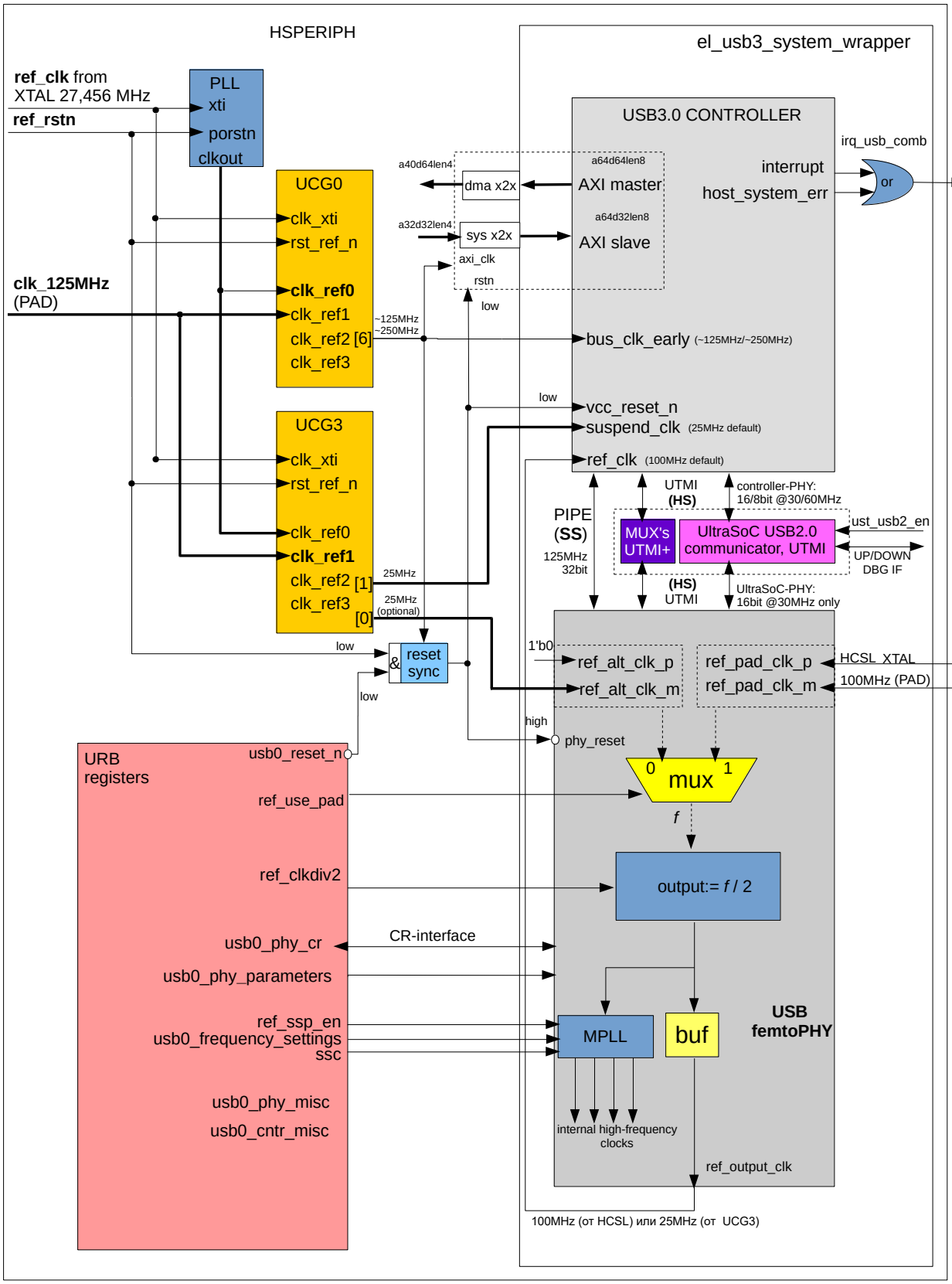


Рисунок 9.8 HSPERIPH. Особенности интеграции USB0

9.9.1 Ограничение адреса DMA

Контроллер USB работает с 64-х разрядным адресом, но при подключении к шине DMA используется только 40 младших разрядов, значения старших игнорируется коммутатором DMA.

9.9.2 Ограничение размера AXI-транзакций DMA

При работе через DMA допускается использовать только транзакции размером до 16 пересылок, т.е. допустимое значение AxLEN от 0 до 15. Необходимо настроить контроллер на работу с AxLEN не больше 15.

9.9.3 Одновременное использование ref_pad_clk и ref_alt_clk

Во избежание перекрёстных помех не допускается одновременное использование тактовых сигналов ref_pad_clk и ref_alt_clk.

9.10 NAND

На следующем рисунке показаны основные особенности интеграции блока NAND:

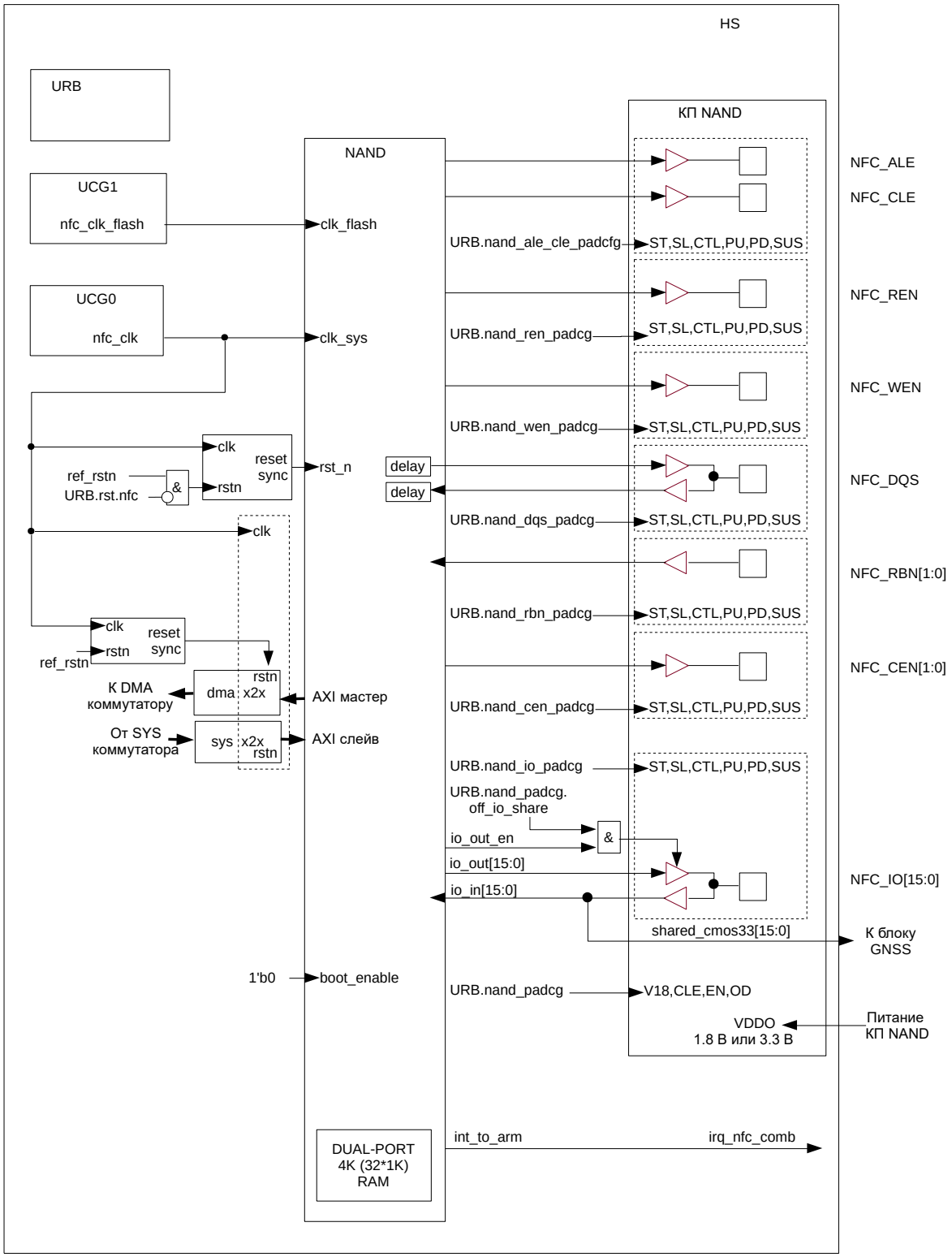


Рисунок 9.9 HSPERIPH. Особенности интеграции NAND

9.10.1 Использование сигнала WP_n

В стандарте ONFI 3.2 описан сигнал WP_n – «Write Protect» (в других версиях стандарта сигнал называется иначе). У контроллера NAND нет выхода WP_n. При необходимости работы с WP_n рекомендуется использовать выводы GPIO.

9.11 SDMMC0, SDMMC1

На следующем рисунке показаны основные особенности интеграции блока SDMMC0 (SDMMC1 аналогично):

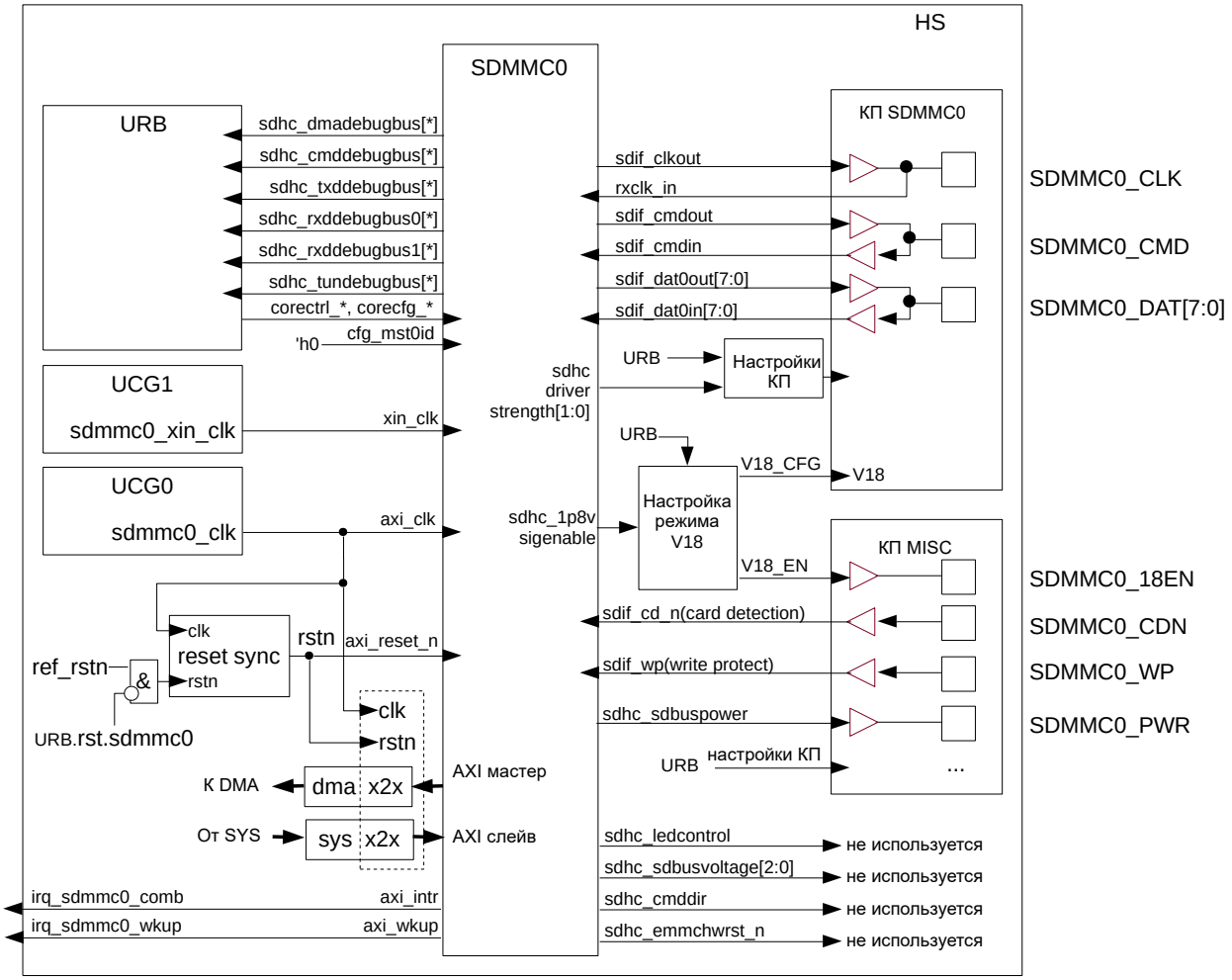


Рисунок 9.10 HSPERIPH. Особенности интеграции SDMMC0

Более детально особенности подключения и настройки КП SDMMC показаны на следующем рисунке:

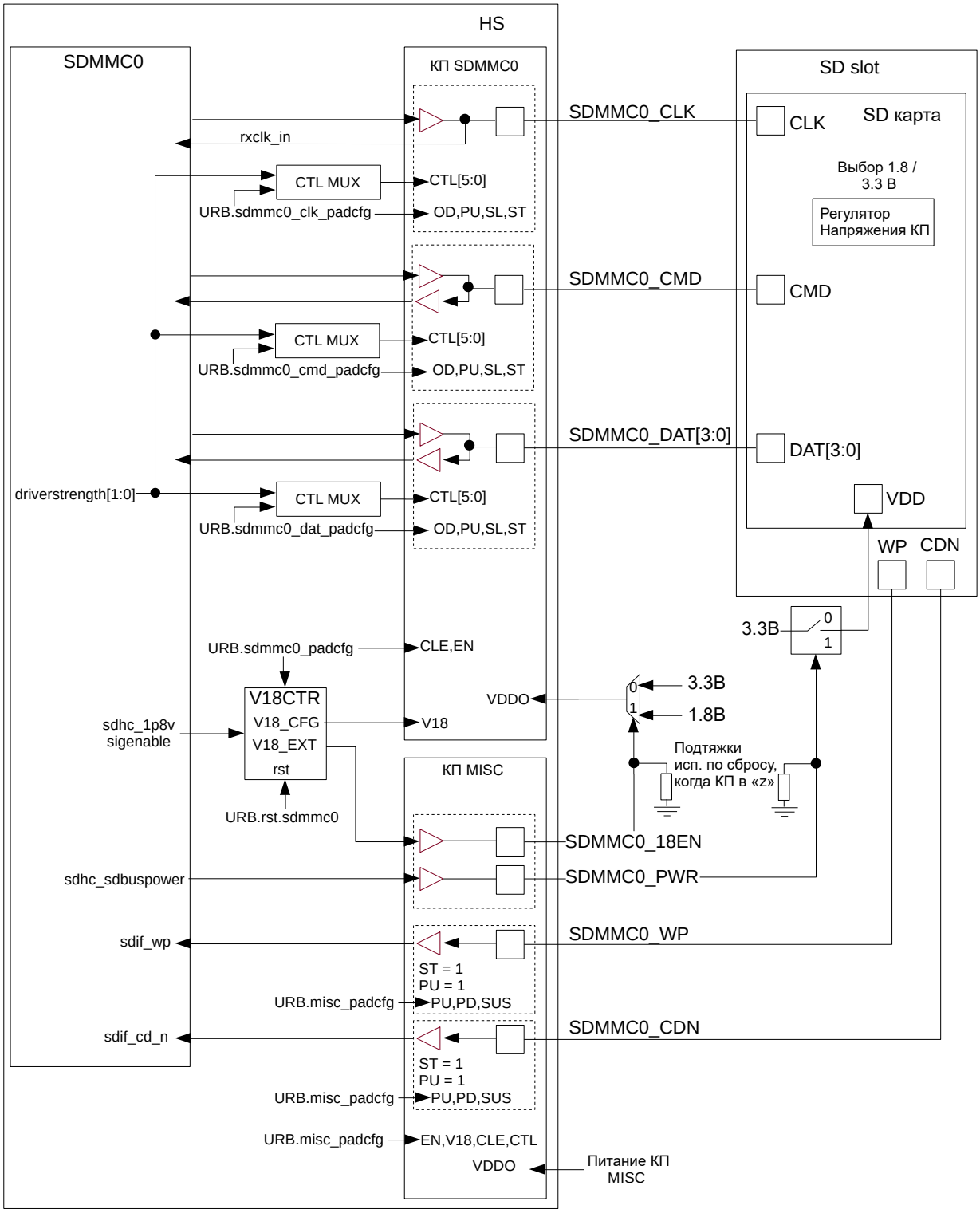


Рисунок 9.11 HSPERIPH. Особенности интеграции SDMMC0 (режим SD)

Сигнальная группа контактных площадок подключается к карте памяти. Управляющая группа подключается к слоту памяти и сигналам управления питанием.

SDMMC*_WP - Запрет записи.
SDMMC*_WP == 0 - write enable

SDMMC*_WP == 1 - write protect
Инверсное состояние входа отображается в регистре "Present State Register (Offset: 0x24)"
Внутри чипа по сбросу подтянут к 1, но можно перенастроить программно.

SDMMC*_CDN - Признак наличия карты.
SDMMC*_CDN == 0 - карта есть
SDMMC*_CDN == 1 - карты нет
Внутри чипа по сбросу подтянут к 1, но можно перенастроить программно.

9.11.1 Настройка типа КП

Таблица 9.119 Соответствие типов КП, входа CTL и сигналов контроллера SDMMC

Тип КП	Вход CTL[5:0] КП	Выход контроллера driverstrength[1:0]
Type A	6'b001111	1
Type B	6'b000111	0
Type C	6'b000011	2
Type D	6'b000001	3

Для настройки типа КП CLK, CMD, DAT используются регистры URB.sdmmc*_clk_padcfg, sdmmc*_cmd_padcfg, sdmmc*_dat_padcfg.

Значение входа CTL КП CLK определяется следующим образом (для CMD и DAT аналогично):

- Если установлен разряд URB.sdmmc*_clk_padcfg.use_sw_ctl, то CTL = URB. sdmmc*_clk_padcfg.ctl, т.е напрямую задаётся из регистра.
- Если не установлен разряд URB.sdmmc*_clk_padcfg.use_sw_ct, то значение CTL определяется значением выхода контроллера driverstrength в соответствии с таблицей выше.

9.11.2 Управление питанием КП

Контактные площади SDMMC разбиты на 3 отдельных независимых группы питания:

- Сигнальные КП SDMMC0 (CLK, CMD, DAT)
- Сигнальные КП SDMMC1 (CLK, CMD, DAT)
- Управляющие КП для SDMMC0 и SDMMC1 (совместно с другими сигналами группы MISC)

Для переключения питания КП CLK, CMD, DAT в контроллере SDMMC предусмотрен выход sdhc_1p8vsigenable. При смене уровня рабочего напряжения необходимо соблюдать последовательности действий определённые ниже. **Внимание! Несоблюдение последовательностей может привести к необратимым повреждениям КП**.

Последовательность переключения из 3.3 В в 1.8 В

1. Сигнал sdhc_1p8vsigenable переводится в значение 1.
2. Внешнее питание КП переключается из 3.3 В в 1.8 В (управляется сигналами SDMMC0_18EN/ SDMMC1_18EN).
3. После того как питание переключилось, необходимо перенастроить КП на работу в режиме 1.8 В. Для этого на входе V18 КП устанавливается 1.
4. Через время ~2 мкс можно начинать обмен данных через КП.

Последовательность переключения из 1.8 В в 3.3 В

1. Сигнал sdhc_1p8vsigenable переводится в значение 0.
2. КП перенастраивается в режим 3.3 В (на входе V18 КП устанавливается 0)
3. Напряжение питания постепенно увеличивается до уровня 3.3 В (~15 мкс). Управляется сигналами SDMMC0_18EN/ SDMMC1_18EN.
4. Через время ~2 мкс после установки нового питания можно начинать обмен данных через КП.

Режимы переключения

Для настройки переключения используются регистры URB.sdmmc0_padcfg и URB.sdmmc1_padcfg. Предусмотрены следующие режимы переключения:

- Программный режим (URB.sdmmc*_padcfg.use_sw_v18 = 1). В этом режиме сигнал контроллера sdhc_1p8vsigenable напрямую подключен к выходу SDMMC*_18EN и управляет внешним переключателем напряжения питания. Настройка V18 КП задается через разряд URB.sdmmc*_padcfg.v18.
- Автоматизированный режим (URB.sdmmc*_padcfg.use_sw_v18 = 0). На следующем рисунке показана временная диаграмма переключения питания в этом режиме:

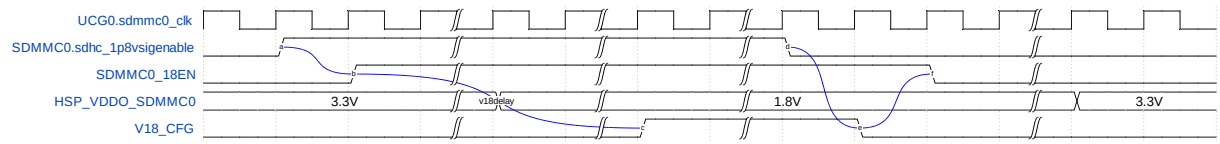


Рисунок 9.12 HSPERIPH. Переключение питания SDMMC0

ПО управляет только сигналом sdhc_1p8vsigenable. Настройка V18 КП при переключении из 3.3 В в 1.8 устанавливается в 1 через время заданное в URB.sdmmc*_padcfg.v18delay.

9.11.3 Последовательность включения КП

В начальный момент времени все КП SDMMC отключены и находятся в третьем состояние. Включение КП SDMMC рекомендуется производить в следующей последовательности:

1. Настроить и включить КП группы MISC URB.misc_padcfg. (К сигналам группы MISC относятся SDMMC*_18EN и SDMMC*_PWR и др.)
2. Дождаться установки стабильного напряжения, управляемого сигналом SDMMC_18EN на входах КП SDMMC.
3. Дождаться отключения питания карты SD, управляемого выводом SDMMC_PWR.
4. Включить остальные КП SDMMC через регистры URB.sdmmc*_padcfg.

9.11.4 Настройка опорной частоты интерфейса SD

Согласно схеме тактирования подсистемы HSPERIPH в качестве опорной частоты интерфейса SD могут выступать частоты CLK125, PLL.clkout, заведенные на делитель UCG1 или частота ref_clk. Значение поля baseclkfreq регистров sdmmc0_corecfg_1, sdmmc1_corecfg_1 должно соответствовать используемой опорной частоте. По сбросу поле baseclkfreq равно 25 - т.е. значение опорной частоты должно быть 25 МГц. Если используется другая частота, необходимо перенастроить baseclkfreq. Например, при работе от ref_clk (в обход PLL и CLK125) и её значении 27.456 МГц необходимо установить baseclkfreq в 27.

9.12 PDMA2

Блок PDMA2 – контроллер DMA общего назначения.

9.12.1 Мастер порты

Для подключения к шинам SYS и DMA в PDMA2 используется 4 мастер интерфейса.

Таблица 9.120 Подключение мастер портов

Мастер порт	Подключение
1	Подключен к коммутатору SYS.
2,3,4	Подключены к коммутатору DMA.

9.12.2 Базовые адресные регистры (БАР)

Мастер порты PDMA2 поддерживают 32-х разрядное адресное пространство. Для работы с 40-разрядным адресом, предусмотрены регистры URB pdma2_bar0, pdma2_bar1, pdma2_bar2. Разряды 39:32 адреса при пересылках через коммутатор DMA напрямую берутся из регистров URB.pdma2_bar*.

32-разрядная шина адреса мастер порта 2 расширяется значением из URB.pdma2_bar0. 32-разрядная шина адреса мастер порта 3 расширяется значением из URB.pdma2_bar1. 32-разрядная шина адреса мастер порта 4 расширяется значением из URB.pdma2_bar2. **Внимание**. Нельзя производить запись в базовые адресные регистры при активности мастер интерфейса. Для изменения настроек необходимо:

1. Дождаться завершения активности мастера.
2. Записать новое значение в нужный БАР.
3. Выдержать паузу достаточную для обновления регистра.
4. Запустить мастер в работу.

9.12.3 Прерывания

Из подсистемы выводится прерывание – логическое или сигналов pdma2_int_combined и| pdma2_hmx_int. pdma2_int_combined_o – комбинационный выход прерывания PDMA2. pdma2_hmx_int – выход прерывания моста AHB-AXI.

9.12.4 Аппаратный интерфейс запросов PDMA2-QSPI1

PDMA2 можно использовать в качестве DMA для QSPI1 (в QSPI1 нет встроенного мастера DMA). При появлении достаточного места в TX FIFO QSPI1 сигнализирует об этом установкой аппаратного сигнала запроса, после чего PDMA2 записывает в него данные. При наличии данных QSPI1 в RX FIFO аппаратный сигнал оповещает об этом PDMA2, после этого PDMA2 вычитывает данные из QSPI. На следующем рисунке показано подключение аппаратного интерфейса запросов PDMA2-QSPI1:

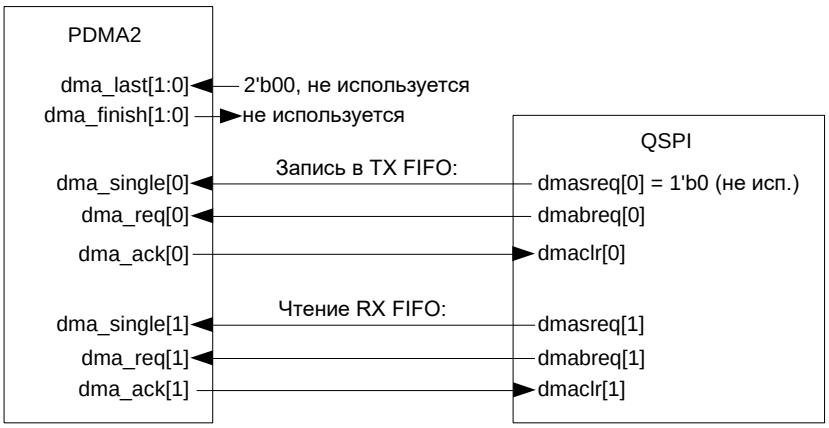


Рисунок 9.13 Интерфейс запросов PDMA2 – QSPI1

9.13 QSPI

Так как блок не имеет встроенного мастера DMA, для обеспечения режима DMA можно использовать блок PDMA2. QSPI подключен к аппаратному интерфейсу запросов PDMA2.

9.13.1 Режимы подключения

Контроллер QSPI позволяет осуществлять пересылки данных в следующих режимах:

- Одноразрядные пересылки (режим norm).
- Двухразрядные пересылки (режим dual)
- Четырёхразрядные пересылки (режим quad)

В зависимости от настроек пересылки производятся либо в режиме full-duplex, либо в режиме half-duplex. В режиме full-duplex данные пересылаются одновременно в двух направлениях. По одному физическому выводу происходит приём, по другому передача данных (сигналы MISO и MOSI). В режиме half-duplex данные передаются по одному и тому же физическому выводу, сначала в одном направлении, затем в другом (SISO).

Настройка режимов производится через регистры CTRL_AUX и GPO_SET.

В следующей таблице показаны допустимые режимы подключения и их настройка.

Таблица 9.121 Настройка режима подключения QSPI1

CTRL_AUX поле qmode	GPO_SET бит 12 (~fdpx)	Режим
2’b00	1	hdpx_norm. Одноразрядная half-duplex пересылка.
2’b01	1	Резерв

CTRL_AUX поле qmode	GPO_SET бит 12 (~fdpx)	Режим
2’b10	1	hdpx_dual. Двухразрядная half-duplex пересылка.
2’b11	1	hdpx_quad. Четырёхразрядная half-duplex пересылка.
2’b00	0	fdpx_norm. Одноразрядная full-duplex пересылка.
2’b01	0	Резерв
2’b10	0	fdpx_dual. Двухразрядная full-duplex пересылка.
2’b11	0	Резерв

Функциональное назначение вывод QSPI отличается в зависимости от режима работы:

Таблица 9.122 Функциональное назначение выводов данных QSPI1

Вывод	Функция в режиме quad half-duplex	Функция в режиме dual half-duplex	Функция в режиме norm half-duplex	Функция в режиме dual full-duplex	Функция в режиме norm full-duplex
QSPI1_SISO3	SISO3	-	-	MISO1	-
QSPI1_SISO2	SISO2	-	-	MOSI1	-
QSPI1_SISO1	SISO1	SISO1	-	MISO0	MISO0
QSPI1_SISO0	SISO0	SISO0	SISO0	MOSI0	MOSI0

9.13.2 GPIO

Предусмотрена возможность использовать любой из выводов QSPI1_SISO* как вывод общего назначения. Это необходимо при подключении к устройствам у которых выводы spi могут иметь дополнительные функции в зависимости от режима работы.

Для управления возможностями GPIO используются регистры GPO_SET и GPO_CLR

Таблица 9.123 Назначение разрядов регистров GPO_SET/GPO_CLR

Разряды GPO_SET /GPO_CLR	Название	Описание
12	fdpx	Включение режима full-duplex. 0 - режим full-duplex включён 1 - режим full-duplex выключен
11:8	gpo_en	Разрешение управления значением выводов QSPI1_SISO* через регистры GPO_SET/GPO_CLR я (акт. уровень – 1)

Разряды GPO_SET /GPO_CLR	Название	Описание
7:4	gpo_oen	Разрешение вывода сигнала QSPI1_SISO* из третьего состояния (акт. уровень – 0).
3:0	gpo	Установка/сброс логического значения QSPI1_SISO*

Считывание текущего значение сигналов QSPI1_SISO* производится через разряды 3:0 регистра GPO_CLR

9.13.3 Особенности мультиплексирования КП

Для поддержки различных режимом подключения (full-duplex/half-duplex) при ограниченном количестве выводов используется мультиплексирование функций некоторых вывод в зависимости от режима. На следующем рисунке отображены основные особенности мультиплексирования (кроме глобальных сигналов включения КП и выбора режима питания 3.3 или 1.8 В):

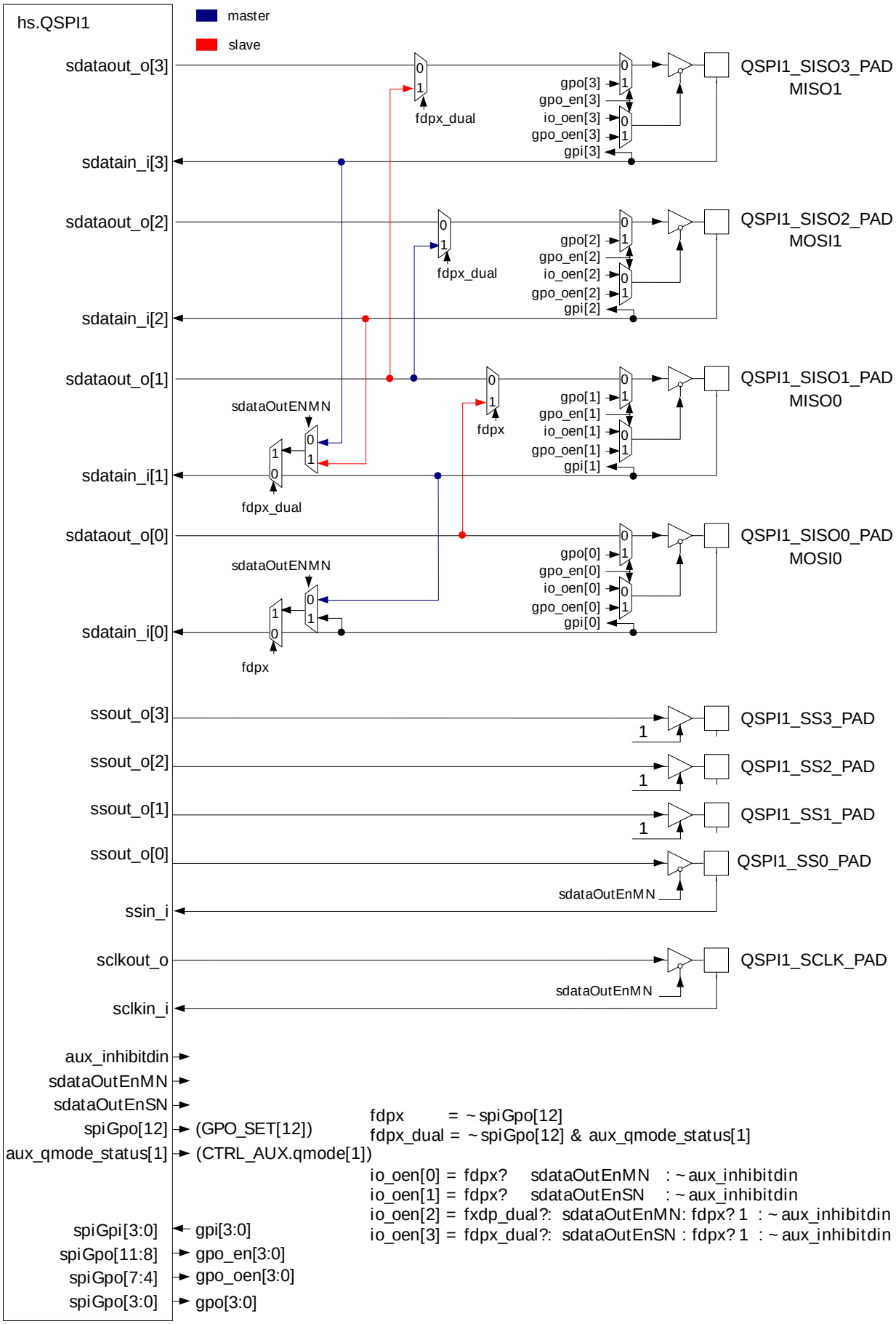


Рисунок 9.14 HSPERIPH. Особенности мультиплексирования КП QSPI

9.13.4XIP

Одновременная работа через XIP и обычный интерфейс не допускается. Сначала необходимо выбрать нужный режим и все пересылки осуществлять только через порт выбранного режима (XIP или не XIP).

Перед включением режима XIP (см. сигналы XIP_EN_REQ/XIP_EN_OUT) необходимо убедиться, что бит enableReq регистра Enable установлен в 0.

Чтение в режиме XIP на границе диапазона адресов

Из-за архитектурных особенностей контроллера QSPI допускается производить чтения только из области адресов ‘h5000_0000 -’h0x5FFF_FBF8 (~32M - 256 байт). Чтения из диапазона ‘h5FFF_FBF0 - ’h5FFF_FFFC категорически запрещены и могут привести к непредсказуемым последствиям.

10. ПОДСИСТЕМА LSPERIPH0

10.1 Общая информация

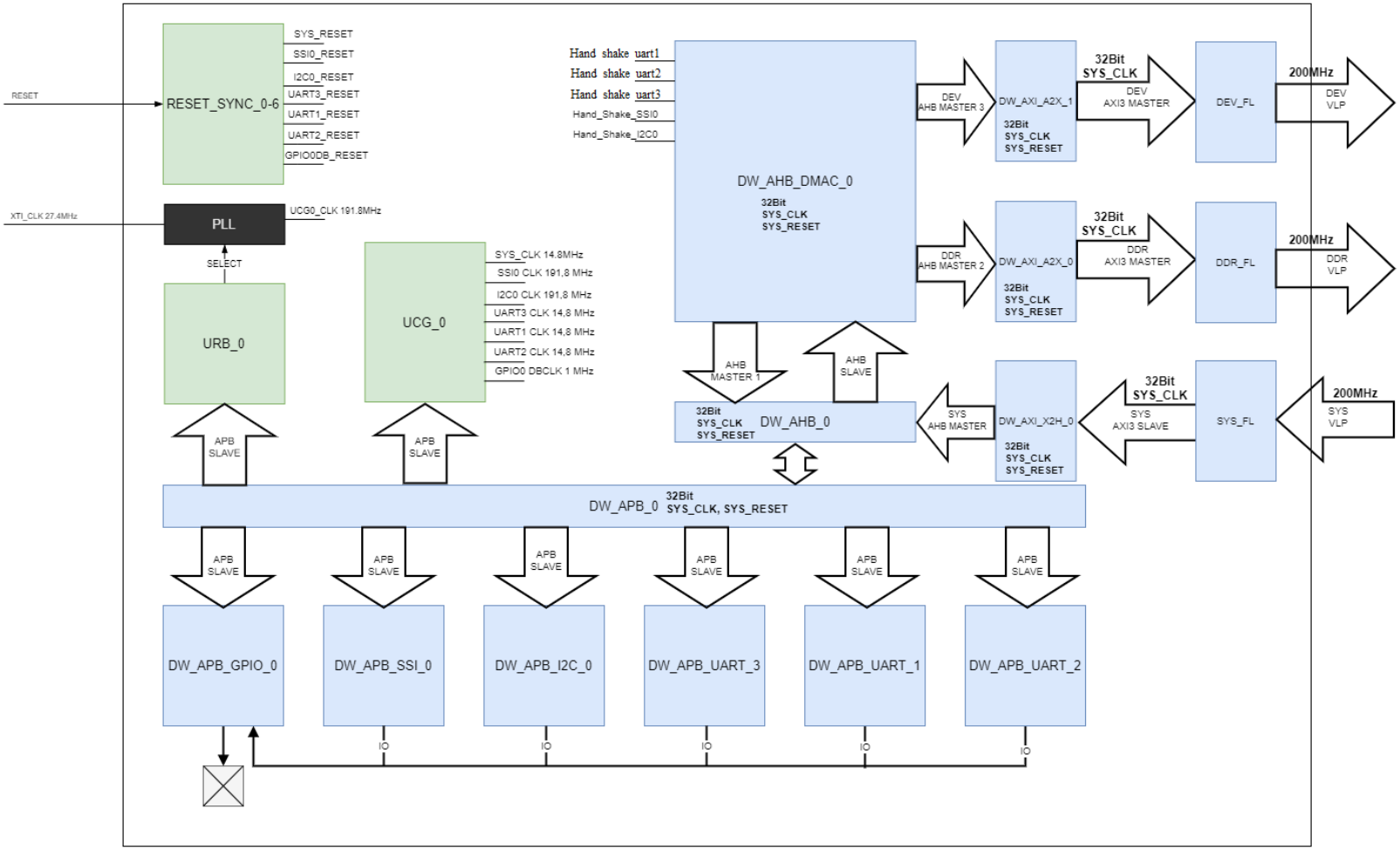


Рисунок 10.1. Структурная схема подсистемы lsperiph0

10.1.1 Состав подсистемы

Подсистема состоит из следующих модулей:

1. URB – универсальный блок регистров для управления подсистемой
2. UCG – блок управления делителями частот подсистемы.
3. GPIO0 – блок интерфейса ввода/вывода общего назначения. Выводы GPIO мультиплексируются с сигналами периферийных интерфейсов.
4. SSI0 – мастер блок управления последовательным периферийным интерфейсом SSI.
5. I2C0 – блок интерфейса подключения к последовательной шине I2C.
6. UART3, UART1, UART2 – блок универсального асинхронного приемо-передатчика.
7. PDMA0 – блок управления прямым доступом к памяти для приема-передачи данных периферийных блоков I2C, SSI, UART. Имеет аппаратный handshake-интерфейс для взаимодействия с этими блоками.
8. PLL – системная PLL.
9. APB, AHB, A2X, X2H – блоки внутрисистемной коммутации.

10.1.2 Адресное пространство

Таблица 10.1. Адресное пространство

Смещение	Конечный адрес	Описание
0x168_0000	0x168_FFFF	Регистры URB
0x169_0000	0x169_FFFF	Регистры UCG2
0x161_0000	0x161_FFFF	Регистры GPIO0
0x162_0000	0x162_FFFF	Регистры SSI0
0x163_0000	0x163_FFFF	Регистры I2C0
0x164_0000	0x164_FFFF	Регистры UART1
0x165_0000	0x165_FFFF	Регистры UART2
0x166_0000	0x166_FFFF	Регистры UART3
0x160_0000	0x160_FFFF	Регистры PDMA0

10.1.3 Прерывания

Таблица 10.2. Прерывания

Прерывание	Источник
Interrupt[7:0]	GPIO0_PORTA
Interrupt[8]	PDMA0
Interrupt[9]	UART3
Interrupt[10]	UART1
Interrupt[11]	UART2
Interrupt[12]	SSI0

Прерывание	Источник
Interrupt[13]	I2C0

Примечание: сброс блока может привести к возникновению фантомного прерывания от SSI.

10.1.4 Прочие сигналы

Таблица 10.3. Прочие сигналы

Название	Назначение	Направление

10.1.5 Подключение интерфейсов квитирования pdma0

Таблица 10.4. Подключение интерфейсов квитирования pdma0

Прерывание	Источник
pdma0_dma_req[0]	i2c0_dma_tx_req
pdma0_dma_req[1]	i2c0_dma_rx_req
pdma0_dma_req[2]	ssi0_dma_tx_req
pdma0_dma_req[3]	ssi0_dma_tx_req
pdma0_dma_req[4]	uart3_dma_tx_req
pdma0_dma_req[5]	uart3_dma_rx_req
pdma0_dma_req[6]	uart1_dma_tx_req
pdma0_dma_req[7]	uart1_dma_rx_req
pdma0_dma_req[8]	uart2_dma_tx_req
pdma0_dma_req[9]	uart2_dma_rx_req

10.1.6 Схема синхронизации

Схема формирования частот периферийных блоков представлена на Рисунок 10.2

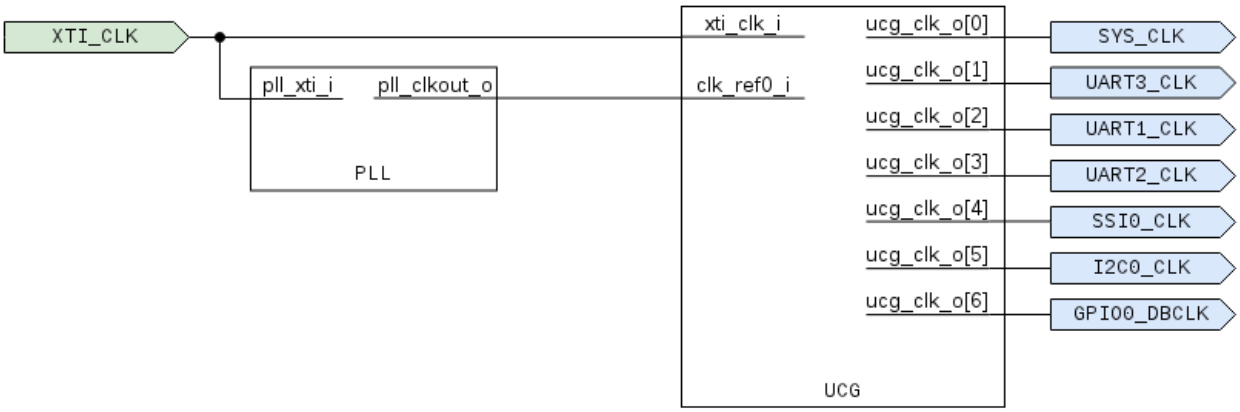


Рисунок 10.2. Схема синхронизации

Таблица 10.5. Требуемые значения частот (коэффициента умножения pll = 7)

Название сигнала	Требуемая частота (МГц)	Делитель UCG
pll_clkout_o	191.8	-
SYS_CLK	191.8	1
UART3_CLK	14,75	13
UART1_CLK	14,75	13
UART2_CLK	14,75	13
SSI0_CLK	191.8	1
I2C0_CLK	191.8	1
GPIO0_DBCLK	1	191

10.2 Регистры

Таблица 10.6. Перечень программно-доступных регистров UCG

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
UCG_CTR0_REG	Регистр управления каналом частоты SYS_CLK	0x00000002	W/R	0x00
UCG_CTR1_REG	Регистр управления каналом частоты UART3_CLK	0x00000001	W/R	0x04
UCG_CTR2_REG	Регистр управления каналом частоты UART1_CLK	0x00000001	W/R	0x08
UCG_CTR3_REG	Регистр управления каналом частоты UART2_CLK	0x00000001	W/R	0x0C
UCG_CTR4_REG	Регистр управления каналом частоты SSI0_CLK	0x00000001	W/R	0x10
UCG_CTR5_REG	Регистр управления каналом частоты I2C0_CLK	0x00000001	W/R	0x14
UCG_CTR6_REG	Регистр управления каналом частоты GPIO_DBCLK	0x00000001	W/R	0x18

Таблица 10.7. Перечень программно-доступных регистров lsperiph0_urb

Смещение	Условное обозначение	Название регистра	Исходное состояние	Тип доступа
0	PLL_CFG	Регистр конфигурации pll	0x0	RW

0x4	PLL_DIAG	Регистр диагностики pll	0x0	RW
0x8	GPIO0_IO_PULL_CTR	Регистр управления подтяжками	0x0	RW

10.2.1 Описание регистра PLL_CFG

Регистр настройки PLL.

Формат регистра PLL_CFG приведен в Таблица 10.8:

Таблица 10.8. Поля регистра PLL_CFG

Диапазон	Название	Описание	Сброс
31	lock	Бит указывает окончание настройки PLL на новую заданную частоту работы. При начале процедуры настройки, либо при выключенной PLL сбрасывается в 0	0
30:27	nr_man	Коэффициент nr настройки pll	0
26:14	nf_man	Коэффициент nf настройки pll	0
13:10	od_man	Коэффициент od настройки pll	0
9	man	При бите заданном в 1 и sel > 0 конфигурация pll определяется полям *_man данного регистра. Выходная частота определяется по формуле ref_clk * (nf_man + 1) / (nr_man + 1) / (od_man + 1). Поля *_man рекомендуется задавать исходя из значений генерируемым скриптом настройки pll. Значение выходной частоты не должно превышать 3600 МГц.	0
8	-	Резерв	0
7:0	sel	Определяет выходную частоту и режим работы PLL: 0x0 - опорная частота ref_clk, PLL выключена; 0x1 - ref_clk * 2; 0x2 - ref_clk * 3; 0x3 - ref_clk * 4; ... - ...; 0x73 - ref_clk * 116; 0x74 - ref_clk * 116; ... - ...; 0xFF - ref_clk * 116; При sel > 0 и man = 1, конфигурация pll определяется полями *_man данного регистра	0

10.2.2 Описание регистра PLL_DIAG

Регистр состояния PLL.

Технологический регистр. Запись в него может привести к неработоспособности подсистемы.

Формат регистра PLL_DIAG приведен в Таблица 10.9:

Таблица 10.9. Поля регистра PLL_DIAG

Диапазон	Название	Описание	Сброс
31:5	-	Резерв	0
4	fbslip	Диагностический выход pll - регистрирует ситуацию feedback cycle slip	0

Диапазон	Название	Описание	Сброс
3	rfslip	Диагностический выход pll - регистрирует ситуацию referency cycle slip	0
2	fasten	Диагностический вход pll - разрешает механизм fast lock	0
1	ensat	Диагностический вход pll - разрешает насыщение, в нормальном режиме - 1	1
0	test	Диагностический вход pll - bypass опорной частоты pll на выход при 1	0

10.2.3 Регистр GPIO0_IOPULL_CTR

Регистр управления резисторной подтяжкой сигналов GPIO0.

Формат регистра GPIO0_IOPULL_CTR приведен в Таблица 10.10.

Таблица 10.10. Формат регистра GPIO0_IOPULL_CTR

Номер бита	Условное обозначение	Назначение
7:0	PORTA_PULL	Каждому биту каждого поля соответствует сигнал управления подтяжкой вверх (pull-up) контактной площадкой GPIO0. 0: подтяжка включена 1: подтяжка выключена
15:8	PORTB_PULL	
23:16	PORTC_PULL	
31:24	PORTD_PULL	

10.3 Функционирование

10.3.1 Особености

Перезапись регистра UCG_CTR0_REG без предварительного включения bypass на ucg может приводить к зависанию подсистемы вплоть до аппаратного сброса.

11. ПОДСИСТЕМА LSPERIPH1

11.1 Общая информация

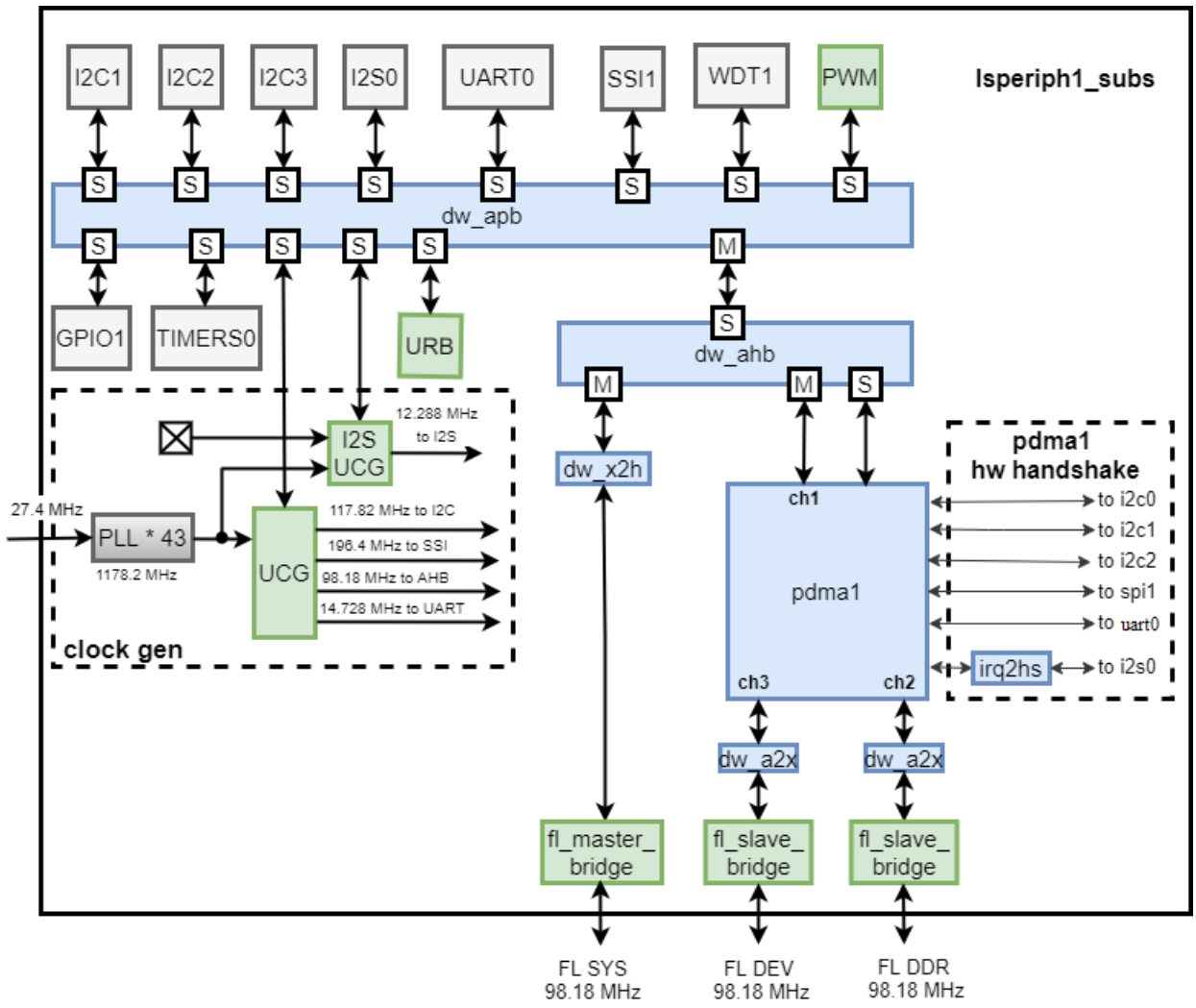


Рисунок 11.1. Структурная схема подсистемы LSPERIPH1

Структурная схема подсистемы представлена на Рисунок 11.1.

11.2 Состав подсистемы

Подсистема состоит из следующих модулей:

1. I2C1, I2C2, I2C3 – блоки интерфейса подключения к последовательной шине I2C.
2. I2S0 - мастер блок управления последовательной шиной I2S, используемой для подключения цифровых аудиоустройств.
3. UART0 – блок универсального асинхронного приемо-передатчика.

4. SSI1 – мастер блок управления последовательным периферийным интерфейсом SSI.
5. GPIO1 – блок интерфейса ввода/вывода общего назначения. Выводы GPIO мультиплексируются с сигналами периферийных интерфейсов и сигналами ШИМ.
6. TIMERS0 – блок универсальных таймеров. Имеет 2 выходных сигнала широтной импульсной модуляции (ШИМ).
7. PWM – дополнительный блок генерации ШИМ сигналов.
8. PDMA1 – блок управления прямым доступом к памяти для приема-передачи данных периферийных блоков I2C, SSI, UART, I2S. Имеет аппаратный handshake-интерфейс для взаимодействия с этими блоками.
9. PLL – системная PLL.
10. UCG – блок управления делителями частот подсистемы.
11. I2S_UCG - отдельный блок управления делителем интерфейсной частоты I2S.
12. WDT1 – сторожевой таймер.
13. URB – универсальных блок регистров. Содержит регистры управления подсистемой.
14. APB, AHB, A2X, X2H – блоки внутрисистемной коммутации.

11.3 Адресное пространство

Таблица 11.1. Адресное пространство

Смещение	Конечный адрес	Описание	Тип доступа
0x0_0000	0x0_FFFF	Регистры PDMA1	none-secure, securable
0x1_0000	0x1_FFFF	Регистры I2C1	
0x2_0000	0x2_FFFF	Регистры I2C2	
0x3_0000	0x3_FFFF	Регистры I2C3	
0x4_0000	0x4_FFFF	Регистры I2S0	
0x5_0000	0x5_FFFF	Регистры UART0	
0x6_0000	0x6_FFFF	Регистры SSI1	
0x7_0000	0x7_FFFF	Регистры PWM	
0x8_0000	0x8_FFFF	Регистры GPIO	
0x9_0000	0x9_FFFF	Регистры TIMERS0	
0xA_0000	0xA_FFFF	Регистры WDT1	
0xB_0000	0xB_FFFF	Резерв	
0xC_0000	0xC_FFFF	Регистры UCG	trusted, securable
0xD_0000	0xD_FFFF	Регистры I2S_UCG	
0xE_0000	0xE_FFFF	Регистры URB	

11.3.1 Матрица коммутации

Таблица 11.2. Матрица коммутации

		Slaves													
		Pdma regs	i2c1 regs	i2c2 regs	i2c3 regs	i2s0 regs	uart0 regs	ssi1 regs	pwm regs	gpio regs	timers regs	wdt1 regs	ucg regs	i2s_ucg regs	urb regs
Masters	Sys port	+	+	+	+	+	+	+	+	+	+	+	+	+	+
	PDMA ch1	+	+	+	+	+	+	+	+	+	+	+			

11.4 Выводы подсистемы

11.4.1 Прерывания

Таблица 11.3. Прерывания

Прерывание	Источник
Interrupts_o [7:0]	GPIO
Interrupts_o [15:8]	TIMERS0
Interrupts_o [16]	I2C1
Interrupts_o [17]	I2C2
Interrupts_o [18]	I2C3
Interrupts_o [19]	SSI1
Interrupts_o [20]	I2S0
Interrupts_o [21]	UART0
Interrupts_o [22]	PWM_INT
Interrupts_o [23]	PWMTU_IN
Interrupts_o [24]	PDMA
Interrupts_o [25]	WDT1

11.4.2 Подключение интерфейсов квитирования pdma1

Таблица 11.4. Подключение интерфейсов квитирования pdma1

Прерывание	Источник
pdma1_dma_req[0]	i2c1_dma_tx_req
pdma1_dma_req[1]	i2c1_dma_rx_req
pdma1_dma_req[2]	i2c2_dma_tx_req
pdma1_dma_req[3]	i2c2_dma_rx_req
pdma1_dma_req[4]	i2c3_dma_tx_req
pdma1_dma_req[5]	i2c3_dma_rx_req
pdma1_dma_req[6]	i2s0_dma_rx0_req
pdma1_dma_req[7]	i2s0_dma_tx0_req
pdma1_dma_req[8]	i2s0_dma_tx1_req
pdma1_dma_req[9]	ssi3_dma_tx_req
pdma1_dma_req[10]	ssi3_dma_tx_req
pdma1_dma_req[11]	uart0_dma_tx_req
pdma1_dma_req[12]	uart0_dma_rx_req

11.5 Схема синхронизации

Схема синхронизации подсистемы представлена на Рисунок 11.2.

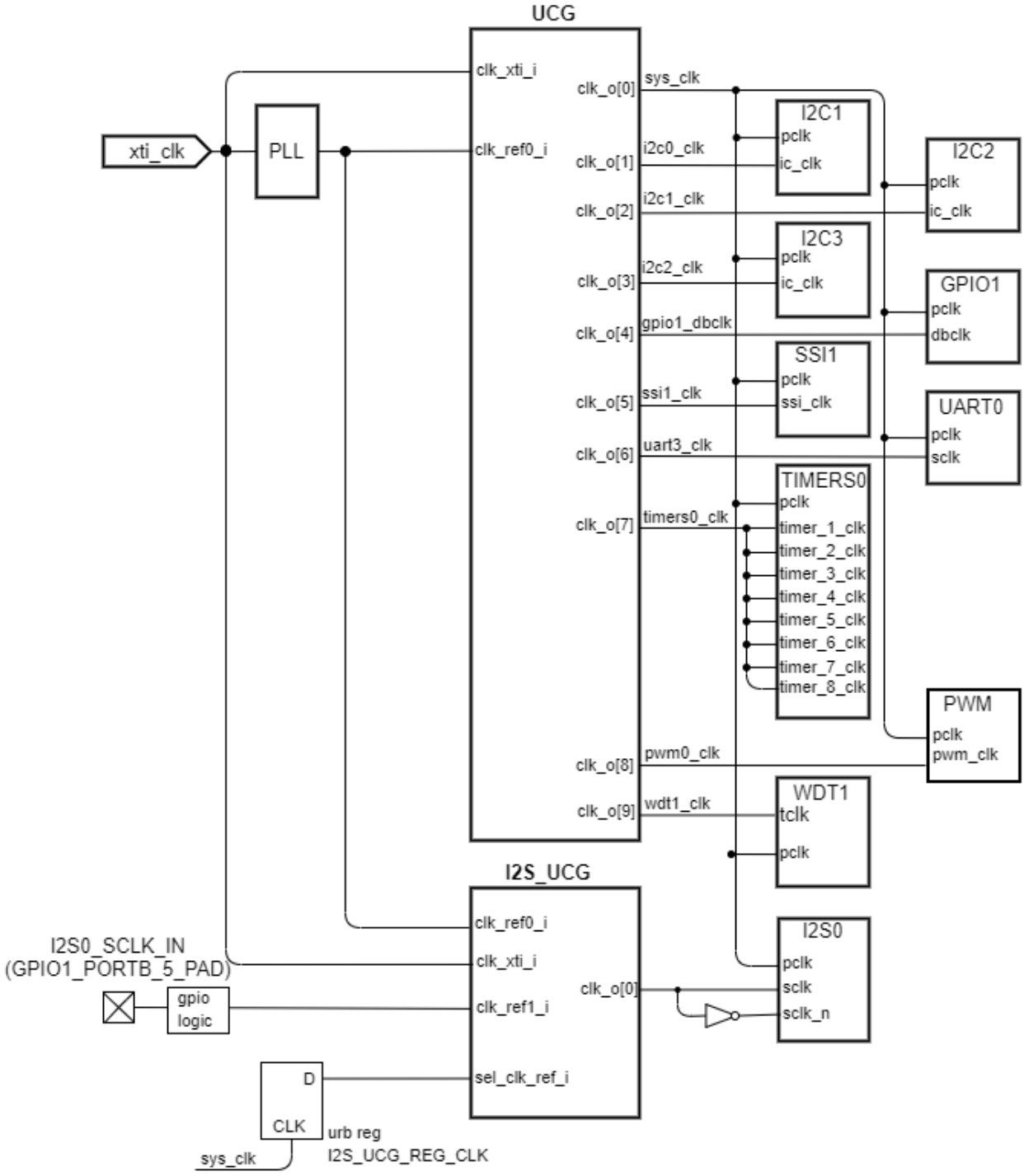


Рисунок 11.2. Схема синхронизации подсистемы

Требуемые значения частот представлены в Таблица 11.5.

Максимальные частоты работы интерфейсов указаны в Таблица 11.6.

Столь высокое требуемое значение выходной частоты PLL обусловлено требованиями к частоте I2S.

Таблица 11.5. Требуемые значения частот подсистемы

Название сигнала	Требуемая частота (МГц)
SYS_CLK	98.18
I2C1_CLK	117.82
I2C2_CLK	117.82
I2C3_CLK	117.82
GPIO1_DB_CLK	98.18
SSI1_CLK	196.4
UART0_CLK	14.728
TIMERS_CLK	98.18
PWM_CLK	98.18
WDT1_CLK	98.18
I2S0_CLK	12.288
PLL	1178.2

Таблица 11.6. Максимальные частоты работы интерфейсов.

Название сигнала	Требуемая частота (МГц)
I2C1	2.95
I2C2	2.95
I2C3	2.95
SSI1	98.2
UART0	14.728
TIMERS	49.9
PWM	49.9
I2S0	12.288

11.6 Регистры

Перечень регистров приведен в Таблица 11.7, Таблица 11.8, Таблица 11.9.

Таблица 11.7. Перечень программно-доступных регистров UCG

Условное Обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
UCG_CTR0_REG	Регистр управления каналом частоты SYS_CLK	0x00000001	W/R	0x00
UCG_CTR1_REG	Регистр управления каналом частоты I2C1_CLK	0x00000000	W/R	0x04
UCG_CTR2_REG	Регистр управления каналом частоты I2C2_CLK	0x00000000	W/R	0x08
UCG_CTR3_REG	Регистр управления каналом частоты I2C3_CLK	0x00000000	W/R	0x0C
UCG_CTR4_REG	Регистр управления каналом частоты GPIO1_DBCLK	0x00000000	W/R	0x10
UCG_CTR5_REG	Регистр управления каналом частоты SSI1_CLK	0x00000000	W/R	0x14
UCG_CTR6_REG	Регистр управления каналом частоты UART3_CLK	0x00000000	W/R	0x18
UCG_CTR7_REG	Регистр управления каналом частоты TIMERS0_CLK	0x00000000	W/R	0x1C
UCG_CTR8_REG	Регистр управления каналом частоты PWM_CLK	0x00000000	W/R	0x20
UCG_CTR9_REG	Регистр управления каналом частоты WDT1_CLK	0x00000000	W/R	0x24

Таблица 11.8. Перечень программно-доступных регистров I2S_UCG

Условное Обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
UCG_CTR0_REG	Регистр управления каналом частоты I2S_CLK.	0x00000000	W/R	0x00

Таблица 11.9. Перечень программно-доступных регистров URB

Условное Обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
PLLCFG	Регистр конфигурации PLL	0x0	W/R	0x00
DIAG	Регистр управления коэффициентом умножения PLL подсистемы.	0x0	W/R	0x04
I2S_UCG_RSTN_PPOLICY	Регистр управления UCG_I2S	0x0	W	0x08
I2S_UCG_RSTN_PSTATUS	Регистр статуса UCG_I2S	0x0	R	0xC
I2S_UCG_REF_CLK	Регистр выбора референсной частоты UCG_I2S	0x0	W/R	0x10
I2C_HIGHSPEED_PULL_ ENABLE	Регистр разрешения аппаратного управления подтяжкой линий SCL I2C интерфейса.	0x0	W/R	0x14
GPIO1_PORTA_0_PAD_CTR	Регистр управления КП GPIO1_PORTA_0_PAD	0xf8	W/R	0x20
GPIO1_PORTA_1_PAD_CTR	Регистр управления КП GPIO1_PORTA_1_PAD	0xf8	W/R	0x24
GPIO1_PORTA_2_PAD_CTR	Регистр управления КП GPIO1_PORTA_2_PAD	0xf8	W/R	0x28
GPIO1_PORTA_3_PAD_CTR	Регистр управления КП GPIO1_PORTA_3_PAD	0xf8	W/R	0x2C
GPIO1_PORTA_4_PAD_CTR	Регистр управления КП GPIO1_PORTA_4_PAD	0xf8	W/R	0x30
GPIO1_PORTA_5_PAD_CTR	Регистр управления КП GPIO1_PORTA_5_PAD	0xf8	W/R	0x34
GPIO1_PORTA_6_PAD_CTR	Регистр управления КП GPIO1_PORTA_6_PAD	0xf8	W/R	0x38
GPIO1_PORTA_7_PAD_CTR	Регистр управления КП GPIO1_PORTA_7_PAD	0xf8	W/R	0x3C
GPIO1_PORTB_0_PAD_CTR	Регистр управления КП GPIO1_PORTB_0_PAD	0xf8	W/R	0x40
GPIO1_PORTB_1_PAD_CTR	Регистр управления КП GPIO1_PORTB_1_PAD	0xf8	W/R	0x44
GPIO1_PORTB_2_PAD_CTR	Регистр управления КП GPIO1_PORTB_2_PAD	0xf8	W/R	0x48
GPIO1_PORTB_3_PAD_CTR	Регистр управления КП GPIO1_PORTB_3_PAD	0xf8	W/R	0x4C
GPIO1_PORTB_4_PAD_CTR	Регистр управления КП GPIO1_PORTB_4_PAD	0xf8	W/R	0x50
GPIO1_PORTB_5_PAD_CTR	Регистр управления КП GPIO1_PORTB_5_PAD	0xf8	W/R	0x54
GPIO1_PORTB_6_PAD_CTR	Регистр управления КП GPIO1_PORTB_6_PAD	0xf8	W/R	0x58
GPIO1_PORTB_7_PAD_CTR	Регистр управления КП GPIO1_PORTB_7_PAD	0xf8	W/R	0x5C
GPIO1_PORTC_0_PAD_CTR	Регистр управления КП GPIO1_PORTC_0_PAD	0xf8	W/R	0x60
GPIO1_PORTC_1_PAD_CTR	Регистр управления КП GPIO1_PORTC_1_PAD	0xf8	W/R	0x64
GPIO1_PORTC_2_PAD_CTR	Регистр управления КП GPIO1_PORTC_2_PAD	0xf8	W/R	0x68

GPIO1_PORTC_3_PAD_CTR	Регистр управления КП GPIO1_PORTC_3_PAD	0xf8	W/R	0x6C
GPIO1_PORTC_4_PAD_CTR	Регистр управления КП GPIO1_PORTC_4_PAD	0xf8	W/R	0x70
GPIO1_PORTC_5_PAD_CTR	Регистр управления КП GPIO1_PORTC_5_PAD	0xf8	W/R	0x74
GPIO1_PORTC_6_PAD_CTR	Регистр управления КП GPIO1_PORTC_6_PAD	0xf8	W/R	0x78
GPIO1_PORTC_7_PAD_CTR	Регистр управления КП GPIO1_PORTC_7_PAD	0xf8	W/R	0x7C
GPIO1_PORTD_0_PAD_CTR	Регистр управления КП GPIO1_PORTD_0_PAD	0xf8	W/R	0x80
GPIO1_PORTD_1_PAD_CTR	Регистр управления КП GPIO1_PORTD_1_PAD	0xf8	W/R	0x84
GPIO1_PORTD_2_PAD_CTR	Регистр управления КП GPIO1_PORTD_2_PAD	0xf8	W/R	0x88
GPIO1_PORTD_3_PAD_CTR	Регистр управления КП GPIO1_PORTD_3_PAD	0xf8	W/R	0x8C
GPIO1_PORTD_4_PAD_CTR	Регистр управления КП GPIO1_PORTD_4_PAD	0xf8	W/R	0x90
GPIO1_PORTD_5_PAD_CTR	Регистр управления КП GPIO1_PORTD_5_PAD	0xf8	W/R	0x94
GPIO1_PORTD_6_PAD_CTR	Регистр управления КП GPIO1_PORTD_6_PAD	0xf8	W/R	0x98
GPIO1_PORTD_7_PAD_CTR	Регистр управления КП GPIO1_PORTD_7_PAD	0xf8	W/R	0x9C
GPIO1_V18	Регистры выбора выходного напряжения всех КП GPIO1	0x0	W/R	0xA0

О регистрах UCG более подробно можно прочитать в Главе 43.

11.6.1 Регистр PLLCFG

Регистр конфигурации PLL.

Формат регистра PLLCFG приведен в следующей таблице:

Таблица 11.10. Поля регистра PLLCFG

Диапазон	Название	Описание	Сброс
7:0	sel	Определяет выходную частоту и режим работы PLL: 0x0 - опорная частота ref_clk, PLL выключена; 0x1 - ref_clk * 2; 0x2 - ref_clk * 3; 0x3 - ref_clk * 4; ... - ...; 0x73 - ref_clk * 116; 0x74 - ref_clk * 116; ... - ...; 0xFF - ref_clk * 116; При sel > 0 и man = 1, конфигурация pll определяется полями *_man данного регистра	
8:8	-	Резерв	0

9:9	man	При бите заданном в 1 и sel > 0 конфигурация pll определяется полям *_man данного регистра. Выходная частота определяется по формуле ref_clk * (nf_man + 1) / (nr_man + 1) / (od_man + 1). Поля *_man рекомендуется задавать исходя из значений генерируемым скриптом настройки pll. Значение выходной частоты не должно превышать 3600 МГц.	0
13:10	od_man	Коэффициент od настройки pll	0
26:14	nf_man	Коэффициент nf настройки pll	0
30:27	nr_man	Коэффициент nr настройки pll	
31	lock	Бит указывает окончание настройки PLL на новую заданную частоту работы. При начале процедуры настройки либо при выключенной PLL сбрасывается в 0.	0

11.6.2 Регистр DIAG

Регистр диагностики PLL.

Формат регистра DIAG приведен в следующей таблице:

Таблица 11.11. Поля регистра DIAG

Диапазон	Название	Описание	Сброс
0:0	test	Диагностический вход pll - bypass опорной частоты pll на выход при 1	0
1:1	ensat	Диагностический вход pll - разрешает насыщение, в нормальном режиме - 1	1
2:2	fasten	Диагностический вход pll - разрешает механизм fast lock	0
3:3	rfslip	Диагностический выход pll - регистрирует ситуацию referency cycle slip	0
4:4	fbslip	Диагностический выход pll - регистрирует ситуацию feedback cycle slip	0
31:5	-	Резерв	

11.6.3 Регистр I2S_UCG_RSTN_PPOLICY

Регистр управления состоянием I2S_UCG.

Регистр используется для управления сбросом I2S_UCG перед изменением источника его референсной частоты через регистр I2S_UCG_REF_CLK. Более подробно см. Главу 44.

Формат регистра I2S_UCG_RSTN_PPOLICY приведен в Таблица 11.12.

Таблица 11.12. Формат регистра I2S_UCG_RSTN_PPOLICY

Номер бита	Условное обозначение	Назначение
4:0	policy	Установка состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; другое - резерв/выключено
31:5	-	Резерв

11.6.4 Регистр I2S_UCG_RSTN_PSTATUS

Регистр статуса состояния I2S_UCG.

Регистр используется для управления сбросом I2S_UCG перед изменением источника его референсной частоты через регистр I2S_UCG_REF_CLK. Более подробно см. Главу 44.

Формат регистра I2S_UCG_RSTN_PSTATUS приведен в Таблица 11.13.

Таблица 11.13. Формат регистра I2S_UCG_RSTN_PSTATUS

Номер бита	Условное обозначение	Назначение
4:0	policy	Поле отражает текущую активную установку состояния: 'b10000 PP_ON - включено; 'b01000 PP_WARM_RST - теплый сброс; другое - резерв/выключено
31:5	-	Резерв

11.6.5 Регистр I2S_UCG_REF_CLK

Регистр выбора источника референсой частоты I2S_UCG.

Формат регистра I2S_UCG_REF_CLK приведен в Таблица 11.14.

Таблица 11.14. Формат регистра I2S_UCG_REF_CLK

Номер бита	Условное обозначение	Назначение
0	I2S_UCG_REF_CLK	Регистр выбора источника референсной частоты I2S_UCG: 0: выход PLL подсистемы. 1: внешний вывод GPIOB[5]. Перед изменением источника необходимо убедиться, что контроллер I2S_UCG находится в состоянии сброса. Управление сбросом I2S_UCG происходит через регистры I2S_UCG_RSTT_PPOLICY и I2S_UCG_RSTN_PSTATUS.
31:1	-	Резерв

11.6.6 Регистр I2C_HIGHSPEED_PULL_ENABLE

Регистр разрешения аппаратного управления подтяжкой линий SCL I2C интерфейса.

Формат регистра I2C_HIGHSPEED_PULL_ENABLE приведен в Таблица 11.15.

Таблица 11.15. Формат регистра I2C_HIGHSPEED_PULL_ENABLE

Номер бита	Условное обозначение	Назначение
0	I2C1_PULL_ENABLE	0: Подтяжкой управляет поле PU регистра GPIO1_PORTA_0_PAD_CTR. 1: Подтяжкой управляет сигнал ic_current_src_en блока i2c1.
1	I2C2_PULL_ENABLE	0: Подтяжкой управляет поле PU регистра GPIO1_PORTA_2_PAD_CTR. 1: Подтяжкой управляет сигнал ic_current_src_en блока i2c2.

Номер бита	Условное обозначение	Назначение
2	I2C3_PULL_ENABLE	0: Подтяжкой управляет поле PU регистра GPIO1_PORTA_4_PAD_CTR. 1: Подтяжкой управляет сигнал ic_current_src_en блока i2c3.
31:3	-	Резерв

11.6.7 Регистр GPIO1_PORT*x*_*n*_PAD_CTR

Регистр управления КП GPIO1_PORT*x*_*n*_PAD, где *x*=A..D, *n*=0..8.

Формат регистра GPIO1_PORT*x*_*n*_PAD_CTR приведен в Таблица 11.16.

Таблица 11.16. Формат регистра GPIO1_PORT*x*_*n*_PAD_CTR

Номер бита	Условное обозначение	Назначение
0	SUS	Pull Keeper. Установка в 1 более чем одного бита {SUS, PU, PD} запрещена.
1	PU	Pull up Установка в 1 более чем одного бита {SUS, PU, PD} запрещена.
2	PD	Pull down Установка в 1 более чем одного бита {SUS, PU, PD} запрещена.
4:3	SL	контроль скорости нарастания. 00 – самая медленная 11 – самая быстрая
10:5	CTL	Регистр управления силой выходного тока КП. Установка в 1 любого бита регистра добавляет 2 мА к силе тока. 000000: z-состояние 000001: 2 mA …… 111111: 12mA
11	-	Резерв
12	E	Включение приемника: 0: Приемник выключен 1: Приемник включен
13	CLE	Включение схемы защиты от повышенного напряжения. Рекомендуется установить в 1 в режиме 3.3В.
14	OD	Перевод КП в режим open-drain
31:15	-	Резерв

11.6.8 Регистр GPIO1_V18

Регистр выбора выходного напряжения всех КП GPIO1

Формат регистра GPIO1_V18 приведен в Таблица 11.17.

Таблица 11.17. Формат регистра GPIO1_V18

Номер бита	Условное обозначение	Назначение
0	V18	Выбор выходного напряжения КП: 0: 3.3 В 1: 1.8 В

Номер бита	Условное обозначение	Назначение
31:1	-	Резерв

12. ЦЕНТРАЛЬНЫЙ ПРОЦЕССОР (CPU)

12.1 Общая информация

Центральный процессор микросхемы представляет из себя четырех ядерный процессор Cortex-A53, являющимся реализацией архитектуры ARMv8-A в энергоэффективном исполнении.

Основные характеристики процессора:

- Реализация архитектуры ARMv8-A включающая:
    - Поддержку режимов исполнения AArch32 и AArch64;
    - Поддержку уровней исполнения EL0, EL1, EL2 и EL3, в каждом из режимов исполнения;
    - Набор инструкций A32;
    - Набор инструкций T32 (Thumb);
    - Набор инструкций A64;
    - Поддержка SIMD и FPU расширений в виде сопроцессора NEON;
- Кэши инструкций и данных L1 по 32 Кбайта, на каждом из ядер;
- Общий разделяемый кэш L2 объемом 1 Мбайт;
- Частота работы – 1.2 ГГц и до 2 ГГц в режиме овердрайв;
- Поддержка контроллера прерываний GIC с версией архитектуры v4;
- Отладка с помощью набора специализированных регистров, адресуемых в общей памяти СнК;
- Порт Accelerator Coherency Port (ACP) с возможность организации IO когерентности с другими блоками СнК.

13. КОНТРОЛЛЕР ПРЕРЫВАНИЙ ЦЕНТРАЛЬНОГО ПРОЦЕССОРА (GIC)

13.1 Общая информация

Контроллер прерываний GIC является реализацией архитектуры ARM Generic Interrupt Controller Architecture версии 3.0 и версии 4.0, совместимой с архитектурой ARMv8 и обслуживает центральный процессор CPU в составе соответствующей подсистемы микросхемы.

Основные характеристики контроллера прерываний:

- Поддержка до 128 физических линий прерываний от устройств микросхемы;
- Выделенный прямой интерфейс AXI4-Stream с центральным процессором;
- Поддержка программно-генерированных прерываний (SGI);
- Поддержка message-based прерываний, реализованных возможностью их записи в блок, с встроенным механизмом трансляции ID (ITS);

14. ГРАФИЧЕСКИЙ ПРОЦЕССОР (GPU)

14.1 Общая информация

Харатериктики:

- Частота обработки 550 МГц;
- 4 shader cores;
- Системный кэш 64K;
- Параллельная обработка множества тайлов;
- Многопоточная Unified Shading Cluster (USC) обработка, включающая pixel shader, vertex shader, GP_GPU (computer shader);
- Высокоэффективная SIMD архитектура ALU;
- Производительность операций с плавающей точкой: 17,5 операций F32/сек, 35 операций F16/сек;
- Геометрическая производительность 0.137 полиномов/сек, текстурная производительность 8,79 текселей/сек, пиксельная производительность 2.19 пикселя/сек;
- Полная совместимость с API OpenGL ES3.2, OpenCL 1.2 EP, Vulkan 1.0, OpenVX 1.x.

Структурная схема GPU представлена на Рисунок 14.1.

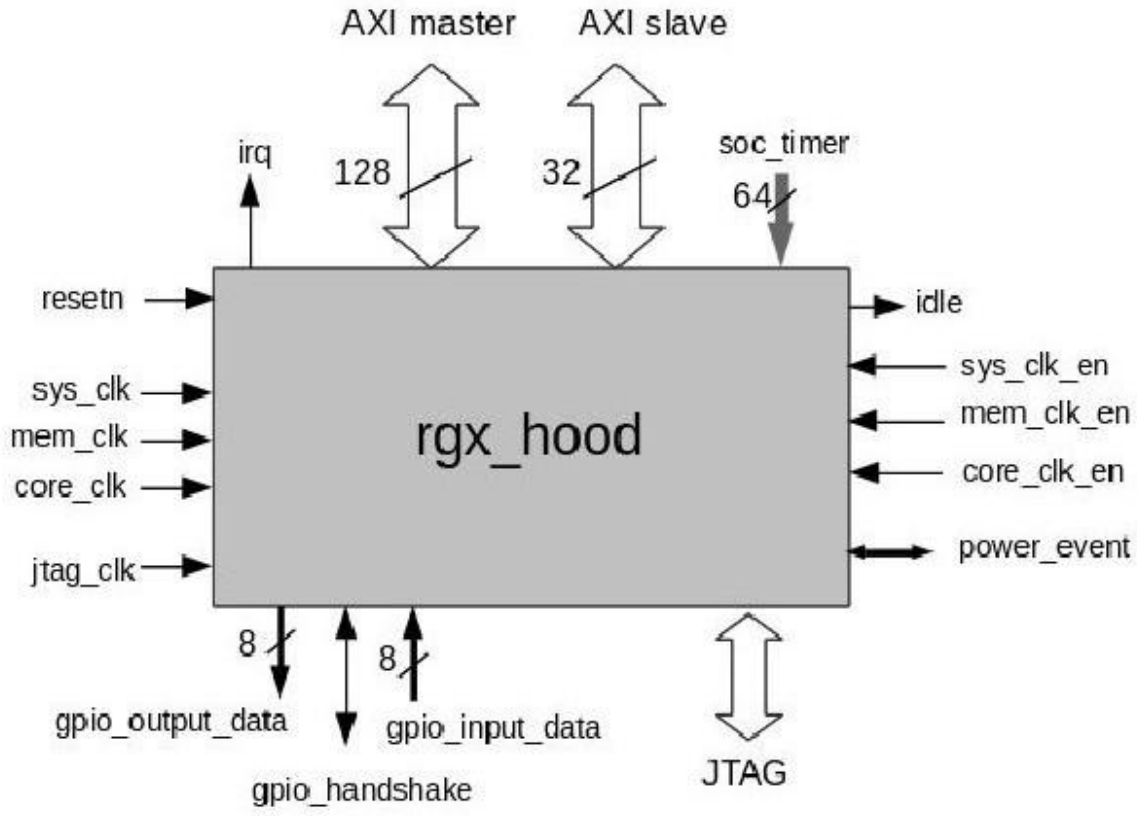


Рисунок 14.1. Структурная схема GPU

Общее регистровое адресное пространство GPU Clark составляет 512KB.

14.2 Конфигурационные параметры

Таблица 14.1. Конфигурационные параметры GPU

Наименование	Описание	Диапазон значений	Значение
AXI_SOCIF_TAG_WIDTH	Определяет ширину ID тэга системного интерфейса в битах (смотри в модуле rgx_hood_axi_pack.vhd)	Min — 4 бита	10
AXI_MEM_USER_WIDTH	Определяет ширину портов AWUSER и ARUSER если таковые используются (смотри в модуле rgx_hood_axi_pack.vhd)		9
TAG_ID_BITS	Через него в том числе определяется значение AXI_MEM_USER_WIDTH (смотри в модуле rgx_slc_cache_pack.vhd)		4

14.3 Интерфейсы блока

14.3.1 Глобальные сигналы

Таблица 14.2. Глобальные сигналы

Наименование	Направление	Описание
sys_clk	in	Синхросигнал системного интерфейса
mem_clk	in	Синхросигнал интерфейса с памятью
core_clk	in	Общий синхросигнал графического ядра

14.3.2 Сигналы сброса

Таблица 14.3. Сигналы сброса

Наименование	Направление	Описание
Resetn	in	Асинхронный системный сброс — активным является низкий уровнь

14.3.3 Сигналы разрешения синхросигналов и общего управления питанием

Таблица 14.4. Сигналы разрешения синхросигналов и общего управления питанием

Наименование	Направление	Описание
sys_clk_en	in	Разрешает включение домена тактового сигнала sys_clk высоким уровнем

mem_clk_en	in	Разрешает включение домена тактового сигнала mem_clk высоким уровнем
core_clk_en	in	Разрешает включение домена тактового сигнала core_clk высоким уровнем
Idle	out	Высоким уровнем на выходе указывает на то что всё графическое ядро находится в «неактивном» состоянии (IDLE state) и соответственно может быть временно полностью отключено
power_event_req	out	Отправка запроса на обработку состояния питания
power_event_type	out	Определяет тип запрашиваемого состояния питания
power_event_complete	in	Ответ о завершении смены состояния питания
power_event_abort	in	Ответ о сбое при смене состояния питания
power_event_domain	out	3-битный выходной порт: является маской, указывающей на индивидуальные домены питания, к которым и применяется вышеупомянутый запрос

14.3.4 Сигналы AXI Slave (AXI3) системного интерфейса

Ниже описан так называемый HOST — SoC interface (SOCIF interface).

Данный системный интерфейс являтеся AXI slave интерфейсом по отношению к самому графическому ядру. Данный интерфейс используется для доступа к контрольным регистрам графического ядра. Ширина шины данных данного интерфейса — строго фиксированна и составляет 32-бита.

Всего в данном интерфейсе 32 порта, что соответствуют стандарту AXI3.

14.3.5 Сигналы AXI Master интерфейса с памятью

Это интерфейс с внешней памятью и соответственно является для графического ядра — AXI master интерфейсом. Содержит в себе канал обозначаемый 'memif0', который используется для чтения и записи данных в память. Ширина шины данных данного интерфейса составляет 128-бит. Преобразование физических адресов, генерируемых с графического ядра на адресный порт, конфигурируется согласно значениям конфигурационных регистров GPU.

Всего в данном интерфейсе максимально предполагается 44 порта в зависимости от конкретной целевой реализации.

14.3.6 Сигналы GPIO

Сигналы GPIO интерфейса предназначены для осуществления взаимодействия с firmware процессором. Данный General Purpose интерфейс был изобретен для серий графических

ядер Series8XE, по которому осуществляется пользовательское взаимодействие между графическим ядром, firmware MIPS процессором, а также какими-либо другими блоками в составе СнК. Данный интерфейс состоит из двух независимых наборов сигналов, каждый набор для своего направления (входной/выходной): «General purpose Input interface» и «General purpose Output interface». Соответственно одно направление определено для запросов, направленных от графического ядра, а другое направление определено для запросов извне к графическому ядру.

Список сигналов верхнего уровня выходного интерфейса General purpose Output interface описаны ниже (здесь запросы отправляет само графическое ядро):

Таблица 14.5. General purpose output interface

Наименование	Направление	Описание
gpio_output_req	out	Выходной сигнал запроса GPIO интерфейса
gpio_output_ack	in	Входной сигнал подтверждения на использование GPIO
gpio_output_data	out	8-разрядные выходные данные с GPIO интерфейса GPU

Список сигналов верхнего уровня входного интерфейса General purpose Input interface описаны ниже (здесь уже запросы отправляются к графическому ядру от внешнего блока):

Таблица 14.6. General purpose input interface

Наименование	Направление	Описание
gpio_input_req	**in**	Входной сигнал запроса на использование GPIO
gpio_input_ack	out	Выходной сигнал подтверждения GPIO
gpio_input_data	**in**	8-разрядные входные данные на GPIO интерфейс GPU

14.3.7 Сигналы конфигурации графического ядра и прочие сигналы

Также, помимо основных вышеуказанных внешних портов и интерфейсов, в графическом ядре предусмотрены сигналы конфигурации и прочие служебные сигналы, приведенные в таблице ниже:

Таблица 14.7. Служебные сигналы

Наименование	Направление	Описание
Irq	**out**	Сигнал прерывания — активный высокий уровень
dxt_bc_enable	in	Этот порт должен быть привязан к статическому значению либо e-fuse. Высокий статический уровень разрешает dxt текстурную компрессию — то есть осуществляет поддержку DirectX при помощи включения блока компрессии текстурных форматов (см. дополнительные условия лицензии).
soc_timer	in	64-разрядный входной порт. Вход для системного таймера. Используется для ввода значения внешнего таймера (используемого для всего SoC) в графическое ядро. Такой внешний таймер обязан быть подключен к данному GPU. Целью такого таймера является прежде всего предоставление возможности для firmware графического ядра поддерживать тот

Наименование	Направление	Описание
		же самый временной ориентир, как и драйверу запущенному на host-CPU. При этом - нет необходимости для firmware графического ядра делать какие-либо дополнительные внутренние подсчеты. Важно то что SoC таймер должен постоянно считать, пока весь SoC включен и даже когда GPU находится в сбросе или отключенном питании. Схема подключения внешнего таймера приведена на рисунке 4. Как на нем показано, входное значение soc_timer на GPU должно быть представленно в Gray coded формате.
astc_enable	in	Этот сигнал имеется у графических ядер, которые поддерживают Khronos Adaptive Scalable Texture Compression (ASTC) стандарт. Необходимо установить в '1' чтобы разрешить ASTC. (см. дополнительные условия лицензии). Этот порт также должен быть привязан к статическому значению либо e-fuse.

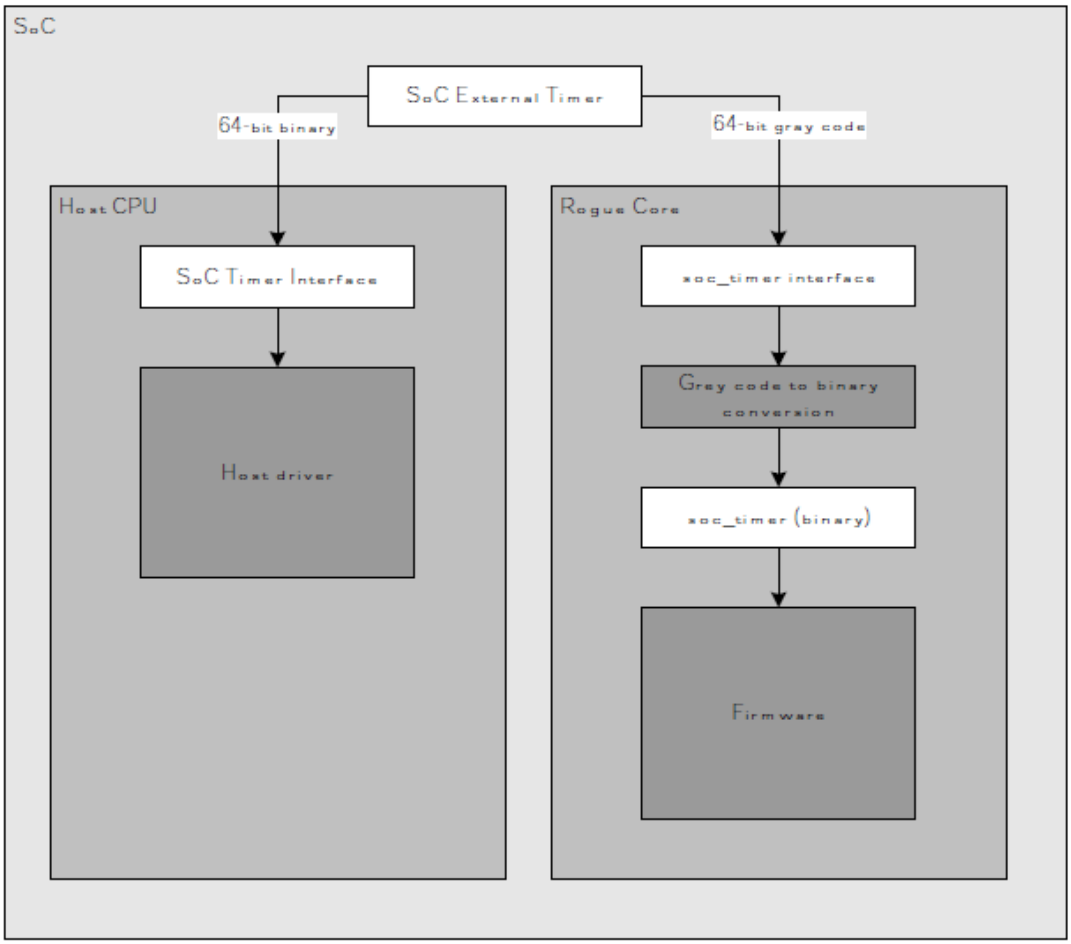


Рисунок 14.2. Схема подключения и работы системного таймера в составе СнК содержащего графическое ядро

14.3.8 Сигналы тестирования и отладки

Данные сигналы будут описаны позже интегратором тестового окружения.

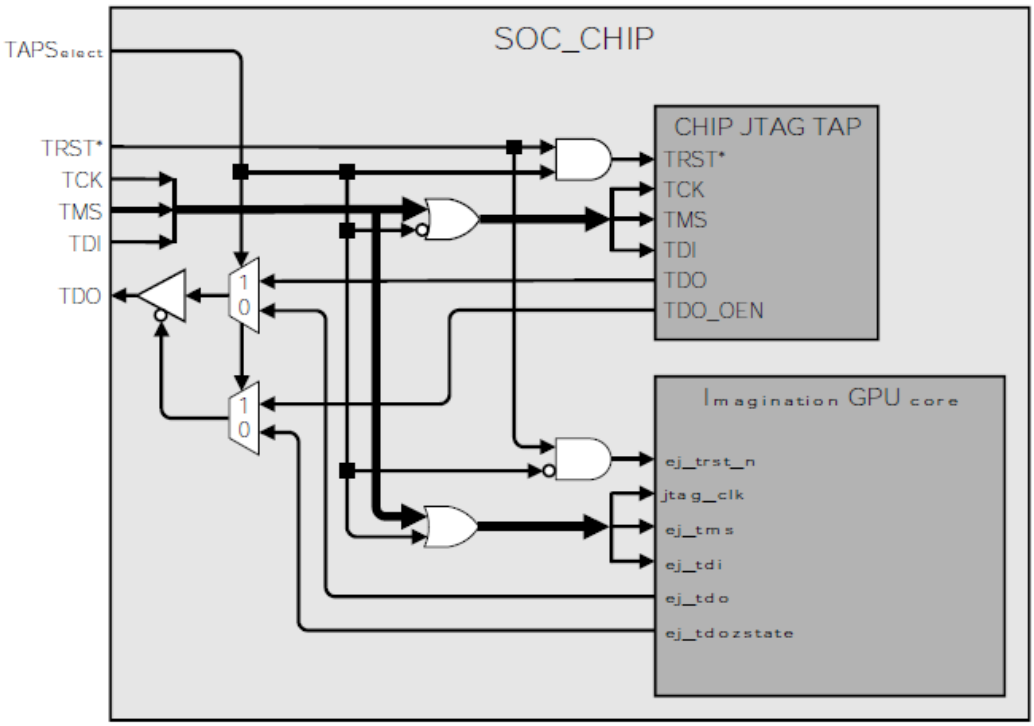


Рисунок 14.3. Схема построения мультиплексирования между TAP-контроллерами JTAG и EJTAG в рамках единой целой СнК

14.3.9 Частоты синхросигналов

Таблица 14.8. Частоты синхросигналов

Наименование синхросигнала	Функциональный диапазон и максимальные значения частот согласно PPA (МГц)	Целевые частоты в рамках подсистемы (МГц)
sys_clk	Any clock frequency;	550
mem_clk	Any clock frequency;	550
core_clk	Any clock frequency;	550

15. БЛОК КОДИРОВАНИЯ И ДЕКОДИРОВАНИЯ ВИДЕО (VPU)

15.1 Общая информация

Кодирование/декодирование видеоданных:

- Поддерживаемые форматы входных видеоданных: RGB 8 бит (RGBA, BGRA, ARGB, ABGR), YUV 420 2/3 плэйна, AFBC YUV422/420, YUV420 10 бит P010, YUV420 10-бит с тайлом 2x2.
- Поддерживаемые стандарты кодирования: HEVC Main/Main10 профили, H.264 Base/Main/High профили, VP8, VP9 0/2 10 бит профили, JPEG.
- Одновременное кодирование и декодирование потоков видео суммарно эквивалентных двум потокам UltraHD 4K 60 fps:
  - два потока кодирования UltraHD 4K 60 fps;
  - кодирование потоков UltraHD 4K 60 fps, FullHD 60 fps, SDTV 30 fps, CIF 30 fps;
  - поток кодирования и поток декодирования UltraHD 4K 60 fps;
  - два потока декодирования UltraHD 4K 60 fps.
- Битрэйт до 400 Мбит/с.
- Частота обработки 600 МГц.

Структурная схема Video processor представлена на Рисунок 15.1. Подробное описание блока содержится в документе «ARM® Mali™-V61 Video Processor. Technical Reference Manual».

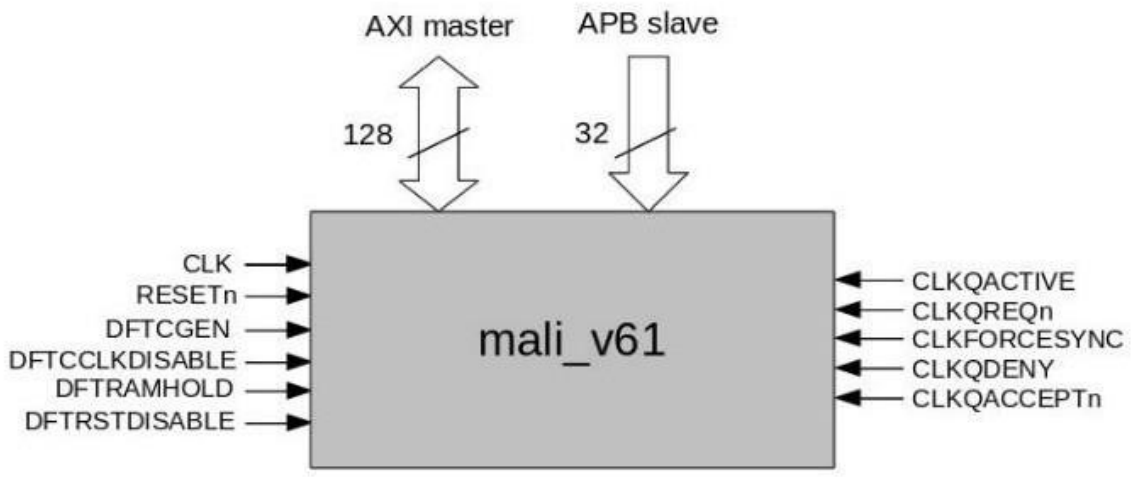


Рисунок 15.1. Структурная схема Mali-V61

15.2 Конфигурационные параметры блока

Таблица 15.1. Конфигурационные параметры Mali-V61

Наименование	Описание	Диапазон значений	Значение
NCORES	Число ядер	1, 2, 3, 4, 6, 8	4
DATA_WIDTH	Число бит для AXI	64, 128	128
CONFIGMASK	Управление функционалом процессора	4095 — наличие всего функционала 4094 — отсутствие кодирования, 3823 — отсутствие поддержки 10 битных пикселей, 3822 — отсутствие кодирования и поддержки 10 битных пикселей, 4031 — только 128-байтные бёрсты пикселей, 4030 — отсутствие кодирования и только 128-байтные бёрсты пикселей, 3759 — отсутствие поддержки 10 битных пикселей и только 128-байтные бёрсты пикселей, 3758 — отсутствие кодирования и поддержки 10 битных пикселей, только 128-байтные бёрсты пикселей	4095

15.3 Интерфейсы блока

15.3.1 Глобальные сигналы

Таблица 15.2. Глобальные сигналы

Наименование	Направление	Описание
CLK	in	Главный синхросигнал
DFTCGEN	in	Разрешение распространения синхросигнала
PRIRESETn	in	Главный асинхронный сигнал сброса
DFTCCLKDISABLE	in	Отключение синхросигнала ядер
DFTRSTDISABLE	in	Разрешение асинхронного сброса внутренних регистров
TSVALUEB	in	Глобальный счётчик временных меток
DFTRAMHOLD	in	Контроль управляющего сигнала для RAM
DFTRSTDISABLE	in	Отключение асинхронного сброса внутренних регистров

15.3.2 Сигналы прерывания

Таблица 15.3. Сигналы прерывания

Наименование	Направление	Описание
IRQ	out	Запрос на прерывание

15.3.3 Сигналы AXI Master

15.3.4 Сигналы APB

15.3.5 Сигналы управления включением/выключением синхросигнала

Таблица 15.4. Сигналы управления включением/выключением синхросигнала

Наименование	Направление	Описание
CLKQACTIVE	out	Включение питания
CLKQREQn	in	Запрос на отключение питания
CLKQACCEPTn	out	Положительный отклик на запрос отключения
CLKQDENY	out	Отрицательный отклик на запрос отключения
CLKFORCESYNC	in	Разрешение синхросигнала для блока питания

15.4 Частоты синхросигналов

Таблица 15.5. Частоты синхросигналов

Наименование синхросигнала	Частота (МГц)
CLK	600

16. ПРОЦЕССОР ПРИЕМА И ПРЕДОБРАБОТКИ ИЗОБРАЖЕНИЯ (ISP)

16.1 Общая информация

- Порты ввода видео:
  - два порта MIPI CSI2 4 лэйна 2500 Мбит/сек
  - один параллельный порт 12 бит до 150 МГц
  - форматы входных видеоданных 2x2 Bayer RGGB, BGGR, GRBG, GBRG
- Предобработка видеоданных «на лету»:
  - 2 контекста видео или статических изображений:
    - при использовании двух контекстов: контекст 0 с максимальным разрешением 4096 x 4486 и контекст 1 с максимальным разрешением 3072 x 4486
    - при использовании одного контекста: контекст 0 с максимальным разрешением 6144 x 4486
  - 2 потока видео UltraHD 4K 30 fps/ 1 поток видео UltraHD 4K 60 fps
  - параллельная обработка двух контекстов
  - одновременная обработка двух пикселей
  - частота обработки 400 МГц
  - разрядность конвейера обработки — 12
  - line-by-line и pixel-by-pixel HDR двух экспозиций
  - frame based HDR программным выполнением HDR merge
  - функции конвейера обработки: коррекция уровня черного, коррекция затенения по краям, коррекция шума, коррекция дефектных пикселей, коррекция хроматических аберраций, преобразование Bayer->RGB, гамма-коррекция, подстройка яркости, контраста, насыщенности, тона, резкости, tone mapping, дробное масштабирование, отображение в цветовое пространство sYCC/sRGB
  - два выходных потока: для видеокодека в формате YCbCr 4:2:2/4:2:0 и для вывода на дисплей в формате RG**B**

Структурная схема ISP представлена на Рисунок 16.1.

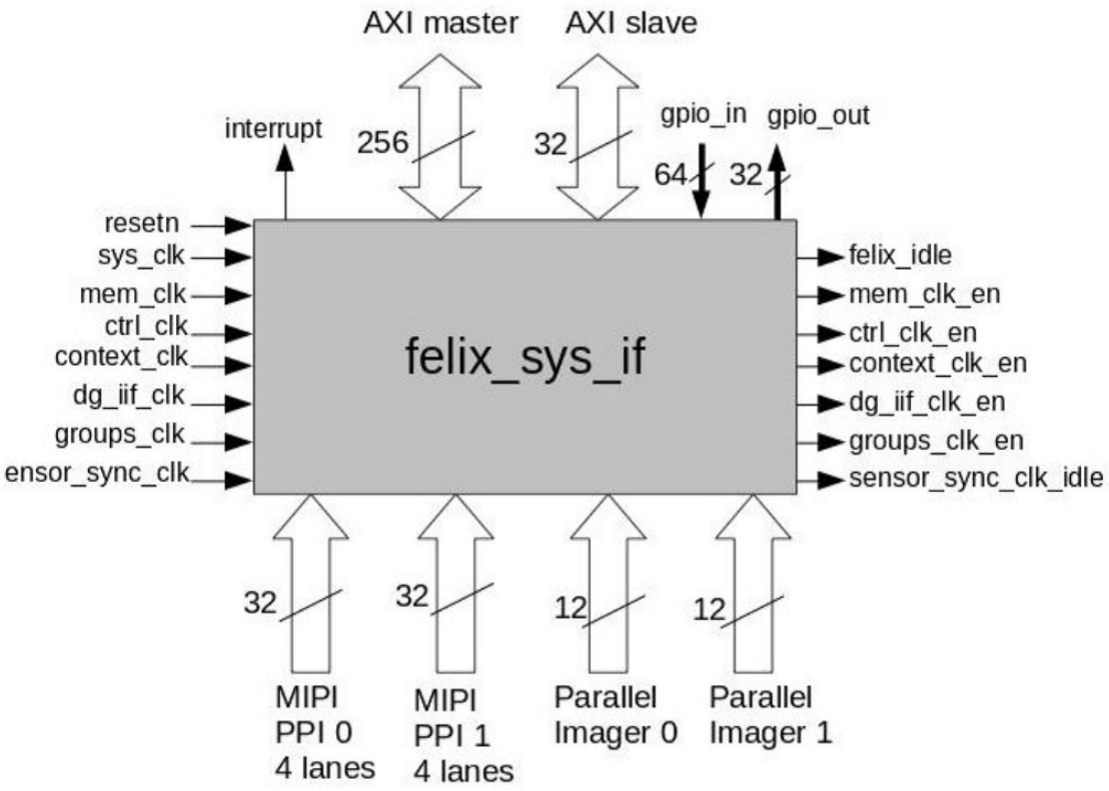


Рисунок 16.1. Структурная схема ISP

16.2 Конфигурационные параметры блока

Таблица 16.1. Конфигурационные параметры V2505

Наименование	Описание		Значение
Параметры интерфейса с сенсорами			
FELIX_NUM_IMAGER	Количество поддерживаемых сенсоров (M)		4
FELIX_GASKET_BIT_DEPTHS[0 to M-1]	Максимальная разрядность данных от сенсоров		12, 12, 12, 12
FELIX_CC_PER_CYCLE	Максимальное количество пикселей на каждый такт sys_clk (L)		2
Параметры внутренней обработки			
DG_IIF_ACTIVE	Наличие внутреннего data generator		1
FELIX_PARALLELISM	Количество пикселей, параллельно обрабатываемых конвейером ISP на один такт (P)		2
FELIX_CONTEXTS	Количество изображений, обрабатываемых в режиме построчного интерливинга (C)		2
FELIX_MAX_CONTEXT_WIDTHS[0 to C-1]	Максимальная ширина изображения для каждого контекста		
		MULT	4096, 3072
		SGL	6144, 3072

Наименование	Описание	Значение
FELIX_MAX_CONTEXT_HEIGHTS[0 to C-1]	Максимальная высота изображения для каждого контекста	4486, 4486
FELIX_MAX_ACTIVE_WIDTH	Максимальная активная ширина контекста 0	6144

16.3 Интерфейсы блока

16.3.1 Глобальные сигналы

Таблица 16.2. Глобальные сигналы

Наименование	Направление	Описание
sys_clk	in	Главный системный синхросигнал
mem_clk	in	Синхросигнал интерфейса с памятью (IMB_BUS4)
Resetn	in	Асинхронный сброс, активный низким уровнем

16.3.2 Синхросигналы обработки данных, синхронные с sys_clk

Таблица 16.3. Синхросигналы обработки данных

Наименование	Направление	Описание
control_clk	in	Синхросигнал внутренней управляющей логики и банка регистров
context_clk[1:0]	in	Синхросигналы приема данных с сенсора через gasket
sensor_sync_clk[3:0]	in	Синхросигнал пересинхронизации с внешнегг сенсора на внетренний домен синхронизации
group_clk[6:0]	in	Синхросигнал для конвейера обработки и блоков управления
dg_if_clk	in	Синхросигнал для Data generator

16.3.3 Сигналы прерывания и статуса

Таблица 16.4. Сигналы прерывания и статуса

Наименование	Направление	Описание
Interrupt	out	Внешний сигнал прерывания
felix_idle	out	Индикатор idle state, может применяться для управления разрешением sys_clk

16.3.4 Сигналы разрешения синхросигналов

Таблица 16.5. Сигналы разрешения синхросигналов

Наименование	Направление	Описание
mem_clk_enable	out	Сигнал разрешения mem_clk
control_clk_enable	out	Сигнал разрешения control_clk

context_clk_enable[1:0]	out	Сигнал разрешения context_clk
sensor_sync_clk_enable[3:0]	out	Сигнал разрешения sensor_sync_clk
group_clk_enable[6:0]	out	Сигнал разрешения group_clk
dg_if_clk_enable	out	Сигнал разрешения dg_if_clk

16.3.5 Сигналы AXI Master

16.3.6 Сигналы AXI Slave

16.3.7 Сигналы интерфейса с памятью

Таблица 16.6. Сигналы интерфейса с памятью

Наименование	Направление	Описание
mem_limited_throughput	in	Сигнал ограничения полосы пропускания памяти для низкоприоритетных запросов

16.3.8 Сигналы MIPI PPI порта сенсоров 0-1

16.3.9 Сигналы параллельного порта сенсоров 2-3

16.3.10 Сигналы индикации

Таблица 16.7. Сигналы индикации

Наименование	Направление	Описание
enc_out_new_line_valid	out	Индикатор записи в память строки
enc_out_frame_start	out	Индикатор начала обработки нового кадра

16.3.11 Сигналы общего назначения

Таблица 16.8. Сигналы общего назначения

Наименование	Направление	Описание
gpio_in[63:0]	In	Входные сигналы общего назначения (по 16 для каждой gasket)
gpio_out[31:0]	Out	Выходные сигналы общего назначения (по 8 для каждой gasket)

16.3.12 Сигналы real-time мониторига

Таблица 16.9. Сигналы real-time мониторига

Наименование	Направление	Описание
rtm_output_a[7:0]	Out	Выходы внутренних статусных сигналов, определяемых регистром RTM_A_CTRL
rtm_output_b[7:0]	Out	Выходы внутренних статусных сигналов, определяемых

		регистром RTM_B_CTRL

16.3.13 Сигналы тестирования и отладки

16.4 Частоты синхросигналов

Таблица 16.10. Частоты синхросигналов

Наименование синхросигнала	Частота (МГц)
sys_clk	400
mem_clk	400

16.5 Блок MC_HDR

16.5.1 Основные особенности

- Формат обрабатываемых видеоданных: RAW Bayer (RGGB, GRBG, GBRG, BGGR)
- Выходной видеопоток 4K 30 fps
- Объединение изображений с двумя экспозициями
- Поддержка различных HDR форматов сенсоров (line by line и pixel by pixel):
    - Staggered HDR (OmniVision)
    - Alternate row HDR (OmniVision)
    - DOL (Sony)
    - Pixel based
- Разрядность входных данных: 8-12
- Разрядность HDR данных: 20
- Глобальный tone mapping с программируемой кривой
- Локальный tone mapping
- Статистика яркости цветовых компонентов: гистограмма по 13 интервалам

16.5.2 Конфигурационные параметры

Аппаратные конфигурационные параметры, определяющие состав и функциональность блока MC_HDR, представлены в Таблица 16.11.

Таблица 16.11. Конфигурационные параметры, определяющие состав и функциональность блока

Параметр	Диапазон значений	Описание
PIXEL_WIDTH_G	12	Разрядность пикселя
CC_PER_CYCLE_G	1-2	Количество пикселей, обрабатываемых за 1 такт

В текущей реализации поддерживается значение PIXEL_WIDTH_G=12.

16.5.3 Перечень сигналов

16.5.3.1 Глобальные сигналы

Таблица 16.12. Глобальные сигналы

Наименование	Направление	Назначение
clk_i	In	Основной синхросигнал блока
ctrl_clk_i	In	Синхросигнал шины управления
resetn_i	in	Сигнал сброса блока (активный низким уровнем), синхронизированный с clk_i. Переход в неактивное состояние (1) должен производиться синхронно с положительным фронтом clk_i (с соблюдением требований recovery/removal) или в отсутствие синхросигнала clk_i

16.5.3.2 Сигналы входного порта видеоданных

Таблица 16.13. Сигналы входного порта видеоданных

Наименование	Направление	Назначение
gasket_valid_i	in	1 — данные на шине gasket_data и сигналы синхронизации на шине gasket_sideband_sync достоверны
gasket_data_i[CC_PER_CYCLE_G*12 — 1:0**]**	in	Входные видеоданные
gasket_sideband_i[2:0**]**	In	Шина сигналов синхронизации. Значения достоверны только при gasket_valid=1 000 — ACTIVE PIXEL 001 — LINE_START 010 — LINE_END 011 — FRAME_START 100 — 111 - резерв

16.5.3.3 Сигналы выходного порта видеоданных

Таблица 16.14. Сигналы выходного порта видеоданных

Наименование	Направление	Назначение
gasket_valid_o	Out	1 — данные на шине gasket_data_o и сигналы синхронизации на шине gasket_sideband_sync_o достоверны
gasket_data_o[CC_PER_CYCLE_G*12 — 1:0**]**	Out	Входные видеоданные

gasket_sideband_o[2:0]	Out	Шина сигналов синхронизации. Значения достоверны только при gasket_valid_o=1 Кодировка аналогична gasket_sideband_sync_o

16.5.3.4 Сигналы порта управления IMG bus.

Таблица 16.15. Сигналы порта управления

Наименование	Направление	Назначение
sys_addr_i [11:0]	in	Шина адреса
sys_wdata_i [31:0]	in	Входная шина записываемых данных
sys_rdnwr_i	in	Направление запроса: 0 — запись 1 — чтение
sys_ad_val_i	in	1 — адрес на шине адреса валиден
sys_ad_en_o	out	1 — запрос разрешен
sys_rdata_o [31:0]	out	Выходная шина считываемых данных
sys_rd_val_o	out	1 — данные на шине sys_rdata валидны

16.5.3.5 Сигналы регистрового интерфейса MC_HDR_CORE

Таблица 16.16. Сигналы регистров управления

Наименование	Направление	Назначение
hdr_mode_i[2:0]	in	Режим HDR
bayer_mode_i[1:0]	in	Маска Байера
line_gap_i[3:0]	in	Количество строк в начале кадра, которое должно быть пропущено (0-15)
frame_ext_first_i	in	Первая строка кадра на входе блока (gasket_sideband_i = FRAME_START): 0 — строка кадра с короткой экспозицией 1 — строка кадра с длинной экспозицией
frame_hsize_i[11:0]	in	Размер кадра по горизонтали
frame_vsize_i[11:0]	in	Размер кадра по вертикали
hdr_enable_i	in	Сигнал разрешения работы блока MC_HDR
tnm_loot_num_i	in	Номер TNM LOOT для чтения
hdr_status_o[2:0]	out	Сигналы состояния блока MC_HDR
black_level_short_i[11:0]	in	Значение уровня черного для кадра с короткой экспозицией
black_level_long_i[11:0]	in	Значение уровня черного для кадра с длинной экспозицией
a_i[15:0]	in	Нормирующий коэффициент для приведения к 20-битным значениям
r_i[15:0]	in	Отношение экспозиций
r_gain_i[12:0]	in	Коэффициент усиления компонента R
gr_gain_i[12:0]	in	Коэффициент усиления компонента GR
gb_gain_i[12:0]	in	Коэффициент усиления компонента GB
b_gain_i[12:0]	in	Коэффициент усиления компонента B
kr_i[11:0]	in	Цветовой коэффициент компонента R
kg_i[11:0]	in	Цветовой коэффициент компонента G

Наименование	Направление	Назначение
kb_i[11:0]	in	Цветовой коэффициент компонента B
hist0_i[23:0] - hist12_i[23:0]	out	Гистограмма

Таблица 16.17. Сигналы интерфейса с блоками памяти

Наименование	Направление	Назначение
mem_addr_i[11:0]	In	Адрес памяти
mem_wr_weight_i	In	Сигнал записи в WEIGHT LOOT
mem_wr_tnm_i	In	Сигнал записи в TNM LOOT
mem_wdata_i[11:0]	In	Входные данные
mem_rd_weight_i	In	Сигнал чтения из WEIGHT LOOT
mem_rd_tnm_i	In	Сигнал чтения из TNM LOOT
mem_rdy_o	Out	Сигнал готовности выходных данных
mem_rdata_o[11:0]	Out	Выходные данные

16.5.4 Функциональная схема

Блок MC_HDR принимает на вход видеоданные в параллельном формате Felix Gasket (см. п. 16.5.5.110). Разрядность данных определяется параметром CC_PER_CYCLE_G (количество пикселей, одновременно обрабатываемых за 1 такт) составляет CC_PER_CYCLE_G*12 разрядов. Одна или две строки входных видеоданных (в зависимости от режима HDR) записываются во входной буфер строк Input Line Buffer (см. Рисунок 16.2). Входной буфер строк осуществляет выравнивание строк с разными экспозициями, и на вход модулей коррекции уровня черного BLC одновременно поступают пиксели с одинаковыми горизонтальными координатами строк изображений с короткой и длинной экспозицией.

После коррекции уровня черного пиксели поступают на вход модуля HDR merge, где выполняется объединение 12-разрядных пикселей с короткой и длинной экспозицией в один 20-разрядный пиксель в соответствии с весовыми коэффициентами, предварительно записанными в Weight LUT. В результате объединения на выход модуля HDR merge выдаются видеоданные с разрядностью CC_PER_CYCLE_G*20.

После коррекции баланса белого в модуле WBC в соответствии с коэффициентами, определенными для каждого типа цветового компонента, данные записываются в HDR Line Buffer, содержащий 2 строки HDR изображения.

Модуль LUM Calc осуществляет вычисление яркости каждого пикселя. Для вычисления 20-битного значения яркости выполняется взвешенное суммирование значений пикселов в его окрестности размером 3x3. Для вычисления требуются пиксели из трех строк одновременно. Пиксели одной строки поступают на вход LUM Calc непосредственно с выхода модуля WB, пиксели двух других строк — с выхода HDR Line Buffer. В результате вычисления определяется 20-разрядное значение яркости L20, которое затем округляется

до 12 бит (L12). Значения L20 и L12 для каждого пикселя поступают на вход модуля Tone Mapper, L12 используется также для вычисления статистики яркости в модуле Statistic.

Модуль Tone Mapper осуществляет глобальное тональное отображение в соответствии со значениями кривой тонального отображения, предварительно записанными в TNM LUT. На выход модуля выдаются видеоданные в параллельном формате Felix Gasket. Разрядность данных составляет CC_PER_CYCLE_G* 12 разрядов.

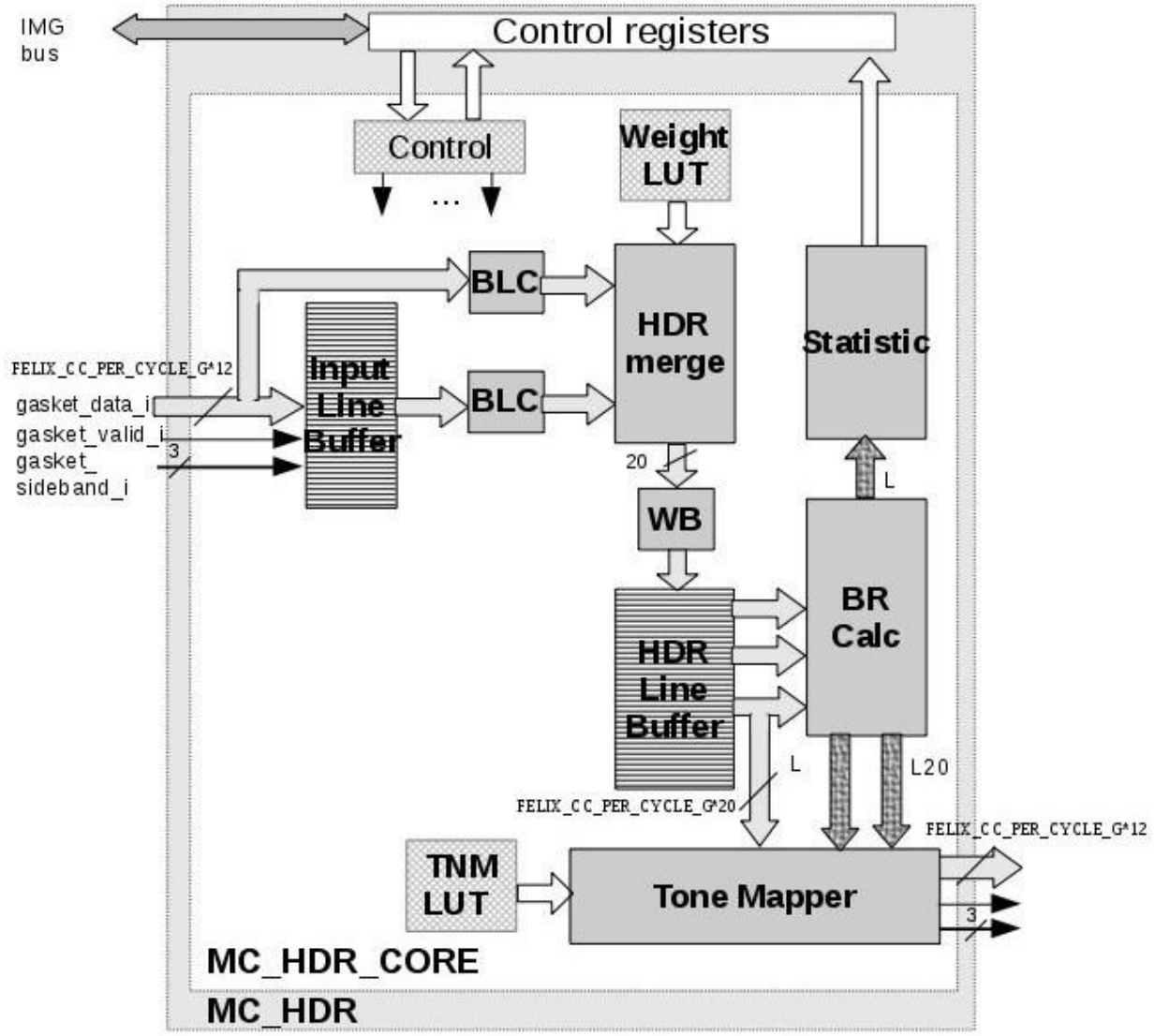


Рисунок 16.2. Функциональная схема блока MC_HDR (PIXEL_WIDTH_G=12)

Модуль Statistic вычисляет значения гистограммы яркости, которые доступны через интерфейс регистров управления.

Модуль Control осуществляет загрузку данных в Weight LUT и TNM LUT, обеспечивает передачу параметров из регистров управления в модули блока.

16.5.5 Функциональное описание

16.5.5.1 Felix Gasket интерфейс

Felix Gasket интерфейс является синхронным параллельным интерфейсом. Данные передаются по шине gasket_data с разрядностью CC_PER_CYCLE_G*12. 3-разрядная шина gasket_sideband используется для передачи управляющей информации о видеопотоке. Значение gasket_sideband=3 сопровождает передачу по шине данные первого пикселя кадра (FRAME_START). Значение gasket_sideband=1 сопровождает передачу по шине данные первого пикселя каждой строки кадра, кроме первой (LINE_START). Значение gasket_sideband_sync=2 сопровождает передачу по шине данные последнего пикселя каждой строки кадра (LINE_END). При передаче остальных пикселей кадра gasket_sideband=0 (ACTIVE PIXEL). Неиспользуемые значения gasket_sideband (4-7) зарезервированы для будущих применений.

Информация на шинах gasket_data и gasket_sideband достоверна только при условии, что сигнал gasket_valid находится в состоянии логической единицы.

Все сигналы Felix Gasket интерфейса синхронны с входным синхросигналом блока clk_i.

На Рисунок 16.3 проиллюстрирована передача первой строки кадра по Felix Gasket интерфейсу. В иллюстрированном примере CC_PER_CYCLE_G = 2, PIXEL_WIDTH_G = 12.

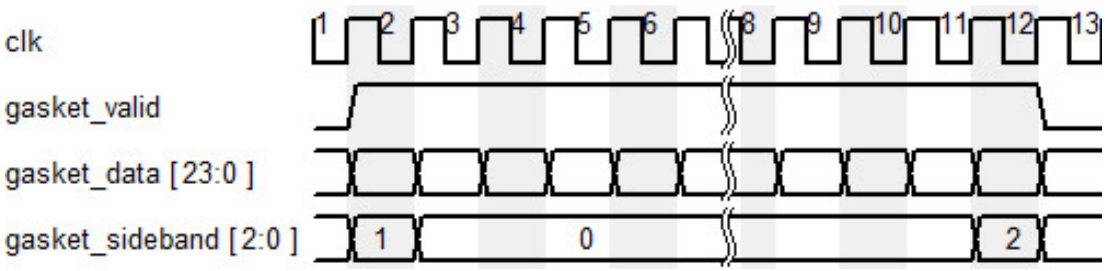


Рисунок 16.3. Передача по Felix Gasket интерфейсу строки кадра, отличной от первой

На Рисунок 16.4 проиллюстрирована передача по Felix Gasket интерфейсу строки кадра, отличной от первой.



Рисунок 16.4. Передача первой строки кадра по Felix Gasket интерфейсу

16.5.5.2 Порт управления (IMG bus)

Блок MC_HDR является slave-устройством на шине управления IMG bus. Через порт управления осуществляется запись в регистры и программируемые LUT и чтение регистров и содержимого программируемых LUT. Интерфейс IMG bus является синхронным, все сигналы порта управления синхронизированы с синхросигналом ctrl_clk_i.

Цикл записи инициируется мастером IMG bus установкой в активное состояние (логической единицы) сигнала sys_ad_val_i. В том же такте на шине адреса sys_addr_i устанавливается адрес записи, а на шине данных sys_wdata_i — записываемые данные (см. Рисунок 16.5. Циклы записи и чтения на IMG bus). Направление передачи данных определяется сигналом sys_rdnwr_i. Цикл записи сопровождается установкой этого сигнала в состояние логического нуля. Для записи одного слова данных по указанному адресу сигнал sys_ad_val_i находится в активном состоянии в течении одного такта. Сигнал разрешения запроса sys_ad_en_o при записи данных всегда находится в активном состоянии (логической единицы), поскольку операция записи выполняется без задержек, поэтому возможна запись нескольких слов данных подряд. В каждом цикле записи (sys_ad_val_i = 1 при sys_rdnwr_i =0) на шинах sys_addr_i и sys_wdata_i должна присутствовать актуальная информация.

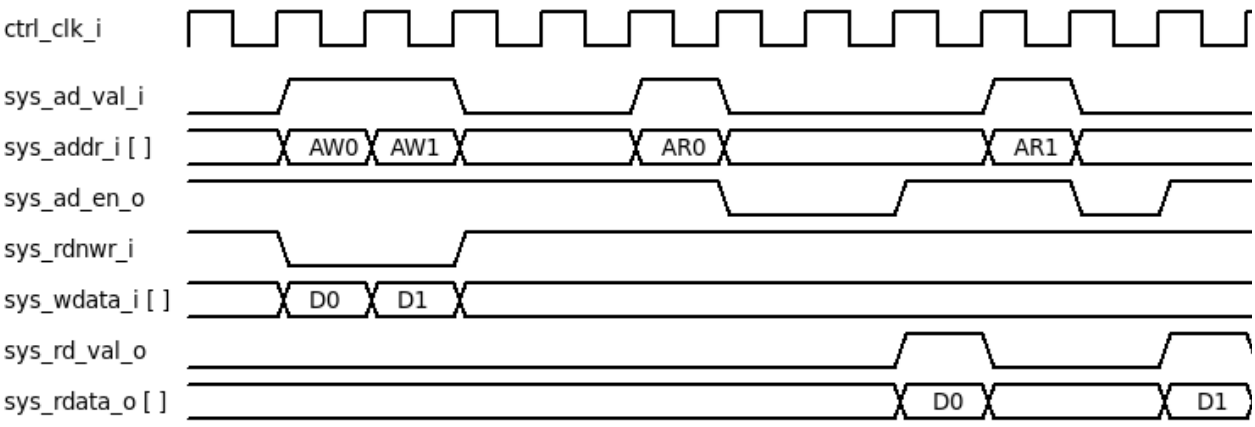


Рисунок 16.5. Циклы записи и чтения на IMG bus

Цикл чтения инициируется мастером IMG bus установкой в активное состояние (логической единицы) сигнала sys_ad_val_i. В том же такте на шине адреса sys_addr_i устанавливается адрес чтения. В случае, если slave не может выдать результат чтения в следующем такте после установки активного sys_ad_val_i, он устанавливает сигнал sys_ad_en_o в состояние логического нуля, сигнализируя мастеру о невозможности выполнить следующий запрос. Когда slave завершает выполнение запроса чтения, он устанавливает в активное состояние (логической единицы) сигнал sys_rd_val_o, сопровождая его выдачей запрашиваемых данных на шину sys_rdata_o. В том же такте, когда устанавливается в активное состояние sys_rd_val_o, сигнал sys_ad_en_o устанавливается в состояние логической единицы, разрешая мастеру выдать следующий запрос на чтение.

16.5.5.3 Регистровый интерфейс модуля MC_HDR_CORE

Регистровый интерфейс обеспечивает передачу информации между блоком регистров и модулем MC_HDR_CORE. Сигналы регистров управления (см. Таблица 16.16) напрямую связывают входы/выходы модуля с соответствующими битами регистров управления (см. п.19). Сигналы интерфейса с блоками памяти обеспечивают доступ к WEIGHT LOOT и TNM LOOT.

Во время цикла записи на шине mem_addr_i устанавливается адрес памяти, на шине mem_wdata_i размещаются данные для записи в память (см. Рисунок 16.6). Для записи в WEIGHT LOOP должен быть установлен в активное состояние (логической единицы) сигнал mem_wr_weight_i. Для записи в TNM LOOP должен быть установлен в активное состояние (логической единицы) сигнал mem_wr_tnm_i. Сигнал записи в память находится в активном состоянии в течение одного такта синхросигнала ctrl_clk. Возможна запись в одну или разные памяти нескольких слов подряд без тактов ожидания.

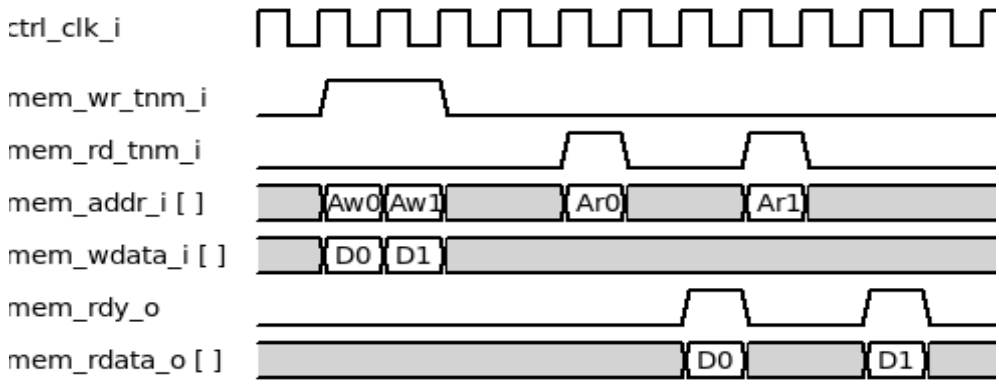


Рисунок 16.6. Интерфейс с блоками памяти

Во время цикла чтения на шине mem_addr_i устанавливается адрес памяти. Для чтения из WEIGHT LOOP должен быть установлен в активное состояние (логической единицы) сигнал mem_rd_weight_i. Для чтения из TNM LOOP должен быть установлен в активное состояние (логической единицы) сигнал mem_rd_tnm_i. Считанные данные появляются на шине mem_rdata_o, сопровождаемые установкой в активное состояние (логической

единицы) сигнала mem_rdy_o. Сигнал находится в активном состоянии в течение одного такта синхросигнала ctrl_clk. Новый цикл чтения из памяти может быть инициирован только после завершения предыдущего, т. е. в следующем такте после установки mem_rdy_o.

16.5.5.4 Входной буфер строк (Input Line Buffer)

Входной буфер строк предназначен для обеспечения одновременной передачи в тракт обработки пикселей с одинаковыми горизонтальными координатами строк кадров с короткой и длинной экспозицией. Количество буферизуемых строк зависит формата передачи HDR данных CMOS сенсором (задается полем HDR_MODE регистра HDR_CONFIG).

В формате staggered HDR или DOL входные данные передаются в следующем порядке: 1 строка кадра с короткой экспозицией, 1 строка кадра с длинной экспозицией и т. д. В этом случае достаточно сохранять в буфере одну строку.

В формате alternate row HDR входные данные передаются в следующем порядке: 2 строки кадра с короткой экспозицией, 2 строки кадра с длинной экспозицией и т. д. В этом случае требуется сохранение в буфере двух строк.

В формате Pixel Based HDR входные данные передаются пиксель за пикселем в одной строке (1 пиксель строки кадра с короткой экспозицией, 1 пиксель строки кадра с длинной экспозицией и т. д). Для этого формата буфер строк не используется, для буферизации пикселя достаточно одного регистра разрядностью 12.

Максимальный размер буфера составляет 2 строки длиной 4K. Организация буфера определяется параметром CC_PER_CYCLE_G. Каждая строка размещается в двух банках BankN_0 и BankN_1 размером 12*CC_PER_CYCLE_G x 2K/CC_PER_CYCLE_G, где N = 0 для четных строк, N=1 для нечетных строк. Пиксели входных данных поочередно записываются в BankN_0 и BankN_1, что обеспечивает возможность одновременных записи новой строки и чтения предыдущей.

16.5.5.5 Коррекция черного (BLC)

Для каждого пикселя выполняется коррекция уровня черного. Два модуля BLC используются для выполнения коррекции черного пикселей строк с короткой и длинной экспозицией. Значения уровня черного для задаются полями BLACK_LEVEL_SHORT BLACK_LEVEL_LONG регистра BLACK_LEVEL_PARAM. BLACK_LEVEL является целым беззнаковым числом в диапазоне от 0 до 4095. Модуль выполняет следующую функцию:

$C_{out} = C$ — BLACK_LEVEL*, где

C — входное значение пикселя,

C_{out} — выходное значение пикселя

Преобразование одинаково для всех типов цветовых компонентов.

16.5.5.6 Объединение строк кадров с различной экспозицией (HDR Merge)

Модуль HDR Merge одновременно принимает на вход CC_PER_CYCLE_G пикселей строк с короткой и длинной экспозицией. Параметры преобразования задаются в регистре HDR_MERGE_PARAM (A — нормирующий коэффициент для приведения к 20-битным значениям, R — отношение экспозиций) и программируемой Weight LUT.

Доступ к Weight LUT осуществляется через регистры WEIGHT_COEFF_CONTROL и WEIGHT_COEFF_DATA. Начальный адрес записи устанавливается в поле COEFF_ADDRESS регистра WEIGHT_COEFF_CONTROL. Далее значения коэффициентов записываются в поле COEFF_DATA регистра WEIGHT_COEFF_DATA. После каждой записи в регистр WEIGHT_COEFF_DATA адрес Weight LUT автоматически инкрементируется аппаратурой. Содержимое Weight LUT доступно для чтения через регистры WEIGHT_COEFF_CONTROL и WEIGHT_COEFF_DATA. Процедура чтения аналогична процедуре записи: устанавливается начальный адрес чтения, затем читается регистр данных и после каждого чтения адрес Weight LUT инкрементируется аппаратурой.

Модуль выполняет следующую функцию:

$C_{out} = A* (C_{long} * Weight[C_{long}] + C_{short} * (1 — Weight[C_{long}])* R)$, где

C_{long} — входное значение пикселя строки с длинной экспозицией,

C_{short} — входное значение пикселя строки с короткой экспозицией,

C_{out} — выходное значение пикселя

Точность и диапазон значений программируемых параметров промежуточных данных приводятся в Таблица 16.18.

Таблица 16.18. Параметры модуля HDR merge

Регистр	Назначение	Разрядность	Формат	Диапазон значений
A	Нормирующий коэффициент для приведения к 20-битным значениям	16	u'8.8	0-256
R	Соотношение экспозиций	16	u'8.8	0-256
WEIGHT[i], i = 0— 4095	Значение весового коэффициента	9	u'1.8	0-1

Преобразование одинаково для всех типов цветовых компонентов.

16.5.5.7 Коррекция баланса белого (WBC)

Коррекция баланса белого осуществляется умножением значения пикселя на коэффициент, соответствующий типу цветового компонента (R, GR, GB, B). Значения коэффициентов для пикселей R и GR задаются полями R_GAIN и GR_GAIN регистра COLOR_COMPONENT0. Значения коэффициентов для пикселей B и GB задаются полями B_GAIN и GB_GAIN

регистра COLOR_COMPONENT1. Значения коэффициентов являются беззнаковым числом с фиксированной точкой (u'3.10) в диапазоне 0-8

16.5.5.8 Буфер строк HDR

Буфер строк HDR предназначен для обеспечения одновременной передачи на модуль вычисления яркости пикселей двух последовательных строк с одинаковыми горизонтальными координатами.

Максимальный размер буфера составляет 2 строки HDR данных длиной 4K. Организация буфера определяется параметром CC_PER_CYCLE_G (см.). Каждая строка размещается в двух банках BankN_0 и BankN_1 размером 20*CC_PER_CYCLE_G x 2K/CC_PER_CYCLE_G, где N = 0 для четных строк, N=1 для нечетных строк. Слова входных данных поочередно записываются в BankN_0 и BankN_1, что обеспечивает возможность одновременных записи новой строки и чтения предыдущей.

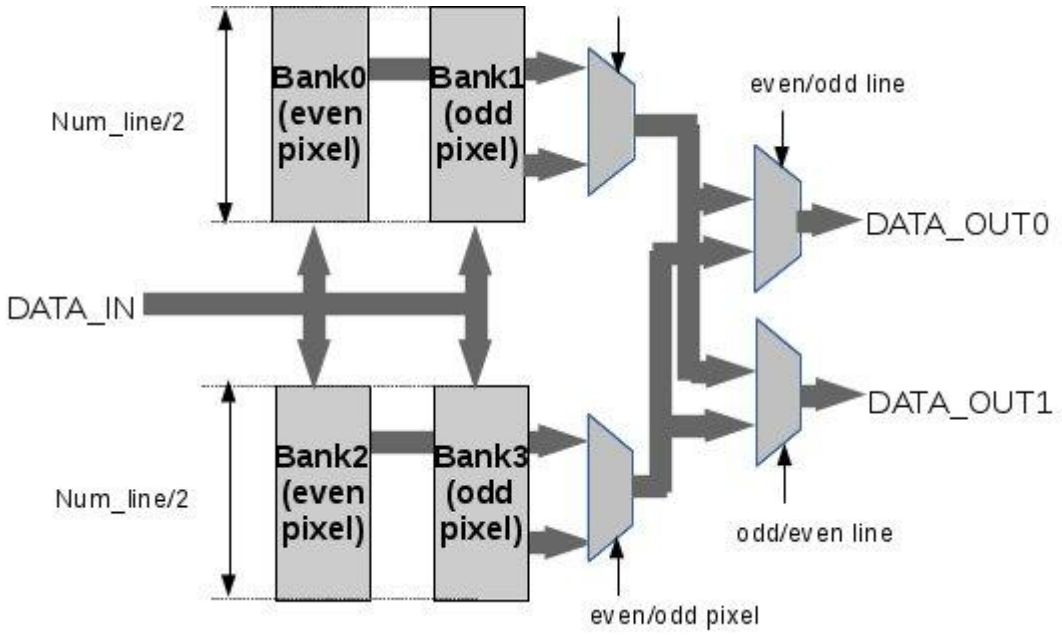


Рисунок 16.7. Структура буфера строк HDR

Запись входных данных DATA_IN осуществляется в один из банков, чтение — из двух банков одновременно (четной и нечетной строки). Если на вход блока поступает четная строка, на выход DATA_OUT0 выдается считанное слово предыдущей нечетной строки (из банка с N=1), на выход DATA_OUT1 - считанное слово предыдущей четной строки (из банка с N=0). Если на вход блока поступает нечетная строка, на выход DATA_OUT0 выдается считанное слово предыдущей четной строки (из банка с N=0), на выход DATA_OUT1 - считанное слово предыдущей нечетной строки (из банка с N=1).

16.5.5.9 Вычисление яркости (LUM Calc)

Для вычисления яркости пикселя используется его окрестность размером 3x3. На вход модуля одновременно поступают 3 слова по CC_PER_CYCLE_G пикселей из трех

последовательных строк: одно с выхода модуля коррекции баланса белого (WBC), два других — из буфера строк HDR. Данные буферизуются в регистрах на входе блока, и при готовности требуемого количества слов для обработки пикселя (окрестности 3x3) выполняются вычисления. Одновременно обрабатываются CC_PER_CYCLE_G пикселей.

20-разрядное значение яркости пикселя с координатами x,y L20 определяется в соответствии со следующей формулой:

$$L20(x,y) = KR * \sum_{\Delta x=\pm1, \Delta y=\pm1} S_R(compBayer(x,y), \Delta x, \Delta y) * C(x+\Delta x, y+\Delta y) + KG * \sum \ldots + KB * \sum \ldots$$

SR представляет собой константную матрицу коэффициентов размерностью 3x3, значения коэффициентов могут принимать значения 0, 1/2, 1/4, или 1/8. Для каждого типа цветового компонента обрабатываемого пикселя compBayer (R, GB, GR, B) определяется своя матрица коэффициентов SR. Для каждого типа compBayer определяются также матрицы коэффициентов SG и SB (см. «Спецификация алгоритма ISP блока MC_HDR»).

Коэффициенты KR, KG, KB задаются в регистрах COLOR_LIGHTNESS_COEFF0-2 соответственно. Значения коэффициентов являются беззнаковыми числами с фиксированной точкой (u'0.12) в диапазоне 0-1.

Вычисленное значение яркости L20 приводится к значению яркости L12 с разрядностью 12 в соответствии с формулой:

$$L12(x,y) = \left(L20(x,y) \gg (\nu - 1)\right) + (\nu - 1) * 2^8,\ \nu = priorEncoder(L20(x,y) \gg 8) \in [0,12], \nu \neq 0$$

В случае ν =0 L12 = L20. Значения L12 находятся в диапазоне от 0 до 3327. На выход модуля выдаются L20 и L12.

16.5.5.10 Тональное отображение (Tone mapper)

Модуль Tone mapper выполняет глобальное тональное отображение 20-разрядного значения пикселя в 12-разрядное значение. 20-разрядные данные поступают на вход модуля с выхода DATA_OUT 0 буфера строк HDR. Значения яркости пикселя L12 и L20 поступают с выхода модуля LUM Calc.

Для всех пикселей кадра используется кривая глобального тонального отображения, загружаемая программно в один из двух банков TNM LUT (0 или 1). Бит TNM_LUT_NUM регистра TNM_LUT_RD_CONTROL выбирает номер TNM LUT, из которого будет производиться чтение кривой. Индексом для TNM LUT является значение яркости L12. Количество слов в TNM LUT составляет 3327. Значения кривой могут находиться в диапазоне 0 до 4095.

Доступ к TNM LUT осуществляется через регистры TONEMAP_CURVE_CONTROL и TONEMAP_CURVE_DATA. Начальный адрес записи устанавливается в поле CURVE_ADDRESS регистра TONEMAP_CURVE_CONTROL. Номер TMN LUT, в который будет производиться запись, определяется битом TNM_LUT_SW_NUM регистра

TONEMAP_CURVE_CONTROL. Далее значения точек кривой записываются в поле CURVE_DATA регистра TONEMAP_CURVE_DATA. После каждой записи в регистр TONEMAP_CURVE_DATA адрес TNM LUT автоматически инкрементируется аппаратурой. Содержимое TNM LUT доступно для чтения через регистры TONEMAP_CURVE_CONTROL и TONEMAP_CURVE_DATA. Процедура чтения аналогична процедуре записи: устанавливается начальный адрес чтения, номер TNM LUT, затем читается регистр данных и после каждого чтения адрес TNM LUT инкрементируется аппаратурой.

Предусмотрено два режима синхронизации процессов записи кривой тонального отображения в TMN LUT и чтения кривой в процессе обработки данных. Режим определяется битом TNM_LUT_NUM_MODE регистра HDR_ENABLE.

В программном режиме синхронизации (TNM_LUT_NUM_MODE=0) переключение бита TNM_LUT_NUM регистра TNM_LUT_RD_CONTROL осуществляется программно при обработке прерывания в конце кадра.

В аппаратном режиме (TNM_LUT_NUM_MODE=1) переключение бита TNM_LUT_NUM регистра TNM_LUT_RD_CONTROL осуществляется аппаратно по окончанию обработки кадра блоком Tone Mapper. После завершения записи кривой тонального отображения в банк, определяемый битом TNM_LUT_SW_NUM регистра TONEMAP_CURVE_CONTROL, должен быть программно установлен бит TNM_LUT_WRITE_END регистра TNM_LUT_STATUS. Когда блок Tone Mapper завершает обработку кадра, анализируется бит TNM_LUT_WRITE_END. При условии TNM_LUT_WRITE_END = 1 бит TNM_LUT_NUM устанавливается аппаратурой равным TNM_LUT_SW_NUM, бит TNM_LUT_WRITE_END аппаратно сбрасывается. Если TNM_LUT_WRITE_END = 0, значение бита TNM_LUT_NUM не изменяется.

Модуль выполняет следующую функцию:

$C_{out} = C * \frac{TNM[L12(x,y)]}{L20(x,y)}$, где

C — входное значение цветового компонента пикселя,

C_{out} — выходное значение цветового компонента

Данные с выхода модуля Tone mapper выдаются на выход блока MC_HDR в соответствии с интерфейсом Felix Gasket.

16.5.5.11 Модуль статистики (Statistics)

Модуль статистики вычисляет гистограмму яркости пикселей по 13 интервалам. Значение яркости L12 поступает с выхода модуля вычисления яркости LUM Calc. Номер интервала, к которому относится значение яркости пикселя L12(x,y) с горизонтальной координатой x и вертикальной координатой y, определяется по формуле:

$\nu = L12(x,y) \gg 8 \in [0,12]$

Алгоритм вычисления гистограммы яркости проиллюстрирован в Таблица 16.19. При поступлении на вход модуля значения яркости для очередного пикселя выполняется инкрементация значения гистограммы одного из 13 интервалов в зависимости от L12.

Таблица 16.19. Алгоритм вычисления гистограммы яркости

L12[11:8]	Действие
0000	HIST0++
0001	HIST1++
0010	HIST2++
0011	HIST3++
0100	HIST4++
0101	HIST5++
0110	HIST6++
0111	HIST7++
1000	HIST8++
1001	HIST9++
1010	HIST10++
1011	HIST11++
1100	HIST12++

Текущее значение гистограммы для интервала i доступно для чтения из регистра HIST$_i$, где i=0-12 в конце кадра и не изменяется в процессе обработки следующего кадра.

16.5.6 Программная модель

16.5.6.1 Программно-доступные регистры

Таблица 16.20. Перечень регистров MC_HDR (0x000 – 0x1FC)

Смещение	Наименование	Доступ	Значение после сброса	Описание
0x000	HDR_CONFIG	RW	0x00000000	
0x004	HDR_FRAME_PARAM	RW	0x00000000	
0x008	HDR_CONTROL	RW	0x00000000	
0x00C	HDR_STATUS	RR	0x00000000	
0x010	BLACK_LEVEL_PARAM	RW	0x00000000	
0x014	HDR_MERGE_PARAM	RW	0x00000000	
0x018	WEIGHT_COEFF_CONTROL	RW	0x00000000	
0x01C	WEIGHT_COEFF_DATA	RW	U	
0x020	COLOR_COMPONENT_CAIN0	RW	0x00000000	
0x024	COLOR_COMPONENT_CAIN1	RW	0x00000000	
0x028	COLOR_LIGHTNESS_COEFF0	RW	0x00000000	
0x02C	COLOR_LIGHTNESS_COEFF1	RW	0x00000000	
0x030-0x03C	Резерв			
0x040	TONEMAP_CURVE_CONTROL	RW	0x00000000	
0x044	TONEMAP_CURVE_DATA	RW	U	
0x48	TONEMAP_LUT_STATUS	R1SC	0x00000000	
0x4C	TONEMAP_LUT_RD_CONTROL	RW	0x00000000	
0x050-0x080	HIST0 — HIST12	RO	0x00000000	
0x094-0x1FC	Резерв			

16.5.6.2 HDR_CONFIG (0x000) Регистр конфигурации блока MC_HDR

Таблица 16.21. Структура регистра HDR_CONFIG

Разряды	Поле	Доступ	Значение после сброса	Описание
2:0	HDR_MODE	RW	0x0	0 — staggered (DOL) HDR 1 — alternate row HDR 2 — pixel HDR 3-7 — резерв
4:3	BAYER_MODE	RW	0x0	Маска Байера: 0 — RGGB 1 — GRBG 2 — GBRG

Разряды	Поле	Доступ	Значение после сброса	Описание
				3 — BGGR
7:5	Резерв			
11:8	LINE_GAP	RW	0x0	Количество строк в начале кадра, которое должно быть пропущено (0-15)
12	FRAME_EXP_FIRST	RW	0	Первая строка кадра на входе блока (gasket_sideband_i = FRAME_START): 0 — строка кадра с короткой экспозицией 1 — строка кадра с длинной экспозицией
31:13	Резерв			

16.5.6.3 HDR_FRAME_PARAM(0x004) Регистр параметров кадра блока MC_HDR

Таблица 16.22. Структура регистра HDR_FRAME_PARAM

Разряды	Поле	Доступ	Значение после сброса	Описание
11:0	FRAME_HSIZE		0x000	Размер выходного кадра по горизонтали Определяется как размер кадра в пикселях
15:12	Резерв			
27:16	FRAME_VSIZE	RW	0x000	Размер выходного кадра по вертикали
31:28	Резерв			

16.5.6.4 HDR_CONTROL (0x008) Регистр управления блока MC_HDR

Таблица 16.23. Структура регистра HDR_CONTROL

Разряды	Поле	Доступ	Значение после сброса	Описание
0	HDR_ENABLE	RW	0	1 — работа блока MC_HDR разрешена
3:1	Резерв			
5	TNM_LUT_NUM_MODE	RW	0	Режим синхронизации записи и чтения кривой тонального отображения 0 – номер банка кривой тонального отображения устанавливается программно 1 – номер банка кривой тонального отображения устанавливается аппаратурой в конце кадра равным значению бита TNM_LUT_SW_NUM регистра TONEMAP_CURVE_CONTROL
31:5	Резерв			

16.5.6.5 HDR_STATUS (0x00C) Регистр состояния блока MC_HDR

Таблица 16.24. Структура регистра HDR_STATUS

Разряды	Поле	Доступ	Значение после сброса	Описание
0	LE_BEFORE_LS	RR	0	1 — признак LINE END получен до получения LINE START
1	LS_BEFORE_LE	RR	0	1 — признак LINE START получен до получения LINE END
2	FS_BEFORE_LE	RR	0	1 — признак FRAME START получен до получения LINE END
31:3	Резерв			

16.5.6.6 BLACK_LEVEL_PARAM (0x010) Регистр параметров коррекции уровня черного

Таблица 16.25. Структура регистра BLACK_LEVEL_PARAM

Разряды	Поле	Доступ	Значение после сброса	Описание
11:0	BLACK_LEVEL_LONG	RW	0x000	Значение уровня черного для кадра с длинной экспозицией
15:12	Резерв			
27:16	BLACK_LEVEL_SHORT	RW	0x000	Значение уровня черного для кадра с короткой экспозицией
31:28	Резерв			

16.5.6.7 HDR_MERGE_PARAM (0x014) Регистр параметров сшивки кадров

Таблица 16.26. Структура регистра HDR_MERGE_PARAM

Разряды	Поле	Доступ	Значение после сброса	Описание
15:0	A	RW	0x0000	Нормирующий коэффициент для приведения к 20-битным значениям
31:16	R	RW	0x0000	Отношение экспозиций

16.5.6.8 WEIGHT_COEFF_CONTROL (0x018). Регистр управления памятью весовых коэффициентов

Таблица 16.27. Структура регистра WEIGHT_COEFF_CONTROL

Разряды	Поле	Доступ	Значение после сброса	Описание
11:0	COEFF_ADDRESS	RW	0x000	Начальное значение адреса загрузки/выгрузки памяти весовых коэффициентов.

Разряды	Поле	Доступ	Значение после сброса	Описание
				Инкрементируется при записи/чтении регистра данных памяти весовых коэффициентов.
31:12	Резерв			

16.5.6.9 WEIGHT_COEFF_DATA (0x01C). Регистр данных памяти весовых коэффициентов

Таблица 16.28. Структура регистра WEIGHT_COEFF_DATA

Разряды	Поле	Доступ	Значение после сброса	Описание
8:0	COEFF_DATA	RW	U	Значение весового коэффициента.
31:9	Резерв			

16.5.6.10 COLOR_COMPONENT_CAIN0 (0x020). Регистр коэффициентов усиления цветовых компонентов 0

Таблица 16.29. Структура регистра COLOR_COMPONENT_GAIN0

Разряды	Поле	Доступ	Значение после сброса	Описание
12:0	R_GAIN	RW	0x000	Значение коэффициента усиления компонента R
15:13	Резерв			
28:16	GR_GAIN	RW	0x000	Значение коэффициента усиления компонента GR
31:29	Резерв			

16.5.6.11 COLOR_COMPONENT_CAIN1 (0x024). Регистр коэффициентов усиления цветовых компонентов 1

Таблица 16.30. Структура регистра COLOR_COMPONENT_GAIN1

Разряды	Поле	Доступ	Значение после сброса	Описание
12:0	GB_GAIN	RW	0x000	Значение коэффициента усиления компонента GB
15:13	Резерв			
28:16	B_GAIN	RW	0x000	Значение коэффициента усиления компонента B
31:29	Резерв			

16.5.6.12 COLOR_LIGHTNESS_COEFF0 (0x028). Регистр коэффициентов яркости цветовых компонентов 0

Таблица 16.31. Структура регистра COLOR_LIGHTNESS_COEFF0

Разряды	Поле	Доступ	Значение после сброса	Описание
11:0	KR	RW	0x000	Значение цветового коэффициента компонента R
15:12	Резерв			
27:16	KG	RW	0x000	Значение цветового коэффициента компонента G
31:28	Резерв			

16.5.6.13 COLOR_LIGHTNESS_COEFF1 (0x02C). Регистр коэффициентов яркости цветовых компонентов 1

Таблица 16.32. Структура регистра COLOR_LIGHTNESS_COEFF1

Разряды	Поле	Доступ	Значение после сброса	Описание
11:0	KB	RW	0x000	Значение цветового коэффициента компонента B
15:12	Резерв			

16.5.6.14 TONEMAP_CURVE_CONTROL (0x040). Регистр управления памятью кривой глобального тонального отображения

Таблица 16.33. Структура регистра TONEMAP_CURVE_CONTROL

Разряды	Поле	Доступ	Значение после сброса	Описание
11:0	CURVE_ADDRESS	RW	0x000	Начальное значение адреса загрузки/выгрузки памяти кривой глобального тонального отображения. Инкрементируется при записи/чтении регистра данных памяти кривой глобального тонального отображения.
12	TNM_LUT_SW_NUM	RW	0	0 – обращение производится к TNM LUT0 1 – обращение производится к TMN LUT1
31:13	Резерв			

16.5.6.15 TONEMAP_CURVE_DATA (0x044). Регистр данных памяти кривой глобального тонального отображения

Таблица 16.34. Структура регистра TONEMAP_CURVE_DATA

Разряды	Поле	Доступ	Значение после сброса	Описание
11:0	CURVE_DATA	RW	U	Значение кривой глобального тонального отображения.
31:12	Резерв			

16.5.6.16 TONEMAP_LUT_STATUS (0x048). Регистр состояния памяти кривой тонального отображения

Таблица 16.35. Структура регистра TONEMAP_LUT_STATUS

Разряды	Поле	Доступ	Значение после сброса	Описание
0	TNM_LUT_WRITE_END	W1SC	0	1 – запись кривой тонального отображения в банк завершена Сбрасывается аппаратурой в конце кадра
31:12	Резерв			

16.5.6.17 TONEMAP_LUT_RD_CONTROL (0x04C). Регистр управления чтением памяти кривой тонального отображения

Таблица 16.36. Структура регистра TONEMAP_LUT_RD_CONTROL

Разряды	Поле	Доступ	Значение после сброса	Описание
0	TNM_LUT_NUM	RW, SCH	0	0 – при работе блока Tone Mapper кривая тонального отображения читается из TNM LOOT0 1 – при работе блока Tone Mapper кривая тонального отображения читается из TNM LOOT1 Если TNM_LUT_NUM_MODE=0, устанавливается программно Если TNM_LUT_NUM_MODE=1, устанавливается аппаратно в конце кадра равным TNM_LUT_SW_NUM
31:12	Резерв			

16.5.6.18 HIST0 (0x050). Регистр гистограммы, интервал 0

Таблица 16.37. Структура регистра HIST0

Разряды	Поле	Доступ	Значение после сброса	Описание
23:0	HIST0	RW	0x000000	Количество пикселей в интервале 0
31:24	Резерв			

16.5.6.19 HIST1_HIST12 (0x054 - 0x080). Регистры гистограммы, интервал 1-12

Аналогично регистру HIST0.

16.6 Интеграция блока MC_HDR

Блок MC_HDR размещается на верхнем уровне иерархии felix_sys_if в тракте данных между felix_gasket_mipi и felix_sys (см. **Рисунок 16.8**). Поскольку текущая конфигурация ISP имеет два порта MIPI, требуется два экземпляра блока MC_HDR: MC_HDR0 и MC_HDR1.

Регистры MC_HDR размещаются в адресном пространстве FELIX_GASKET:

0xD000 - 0xD1FF MC_HDR0
0xD200 - 0xD3FF MC_HDR1

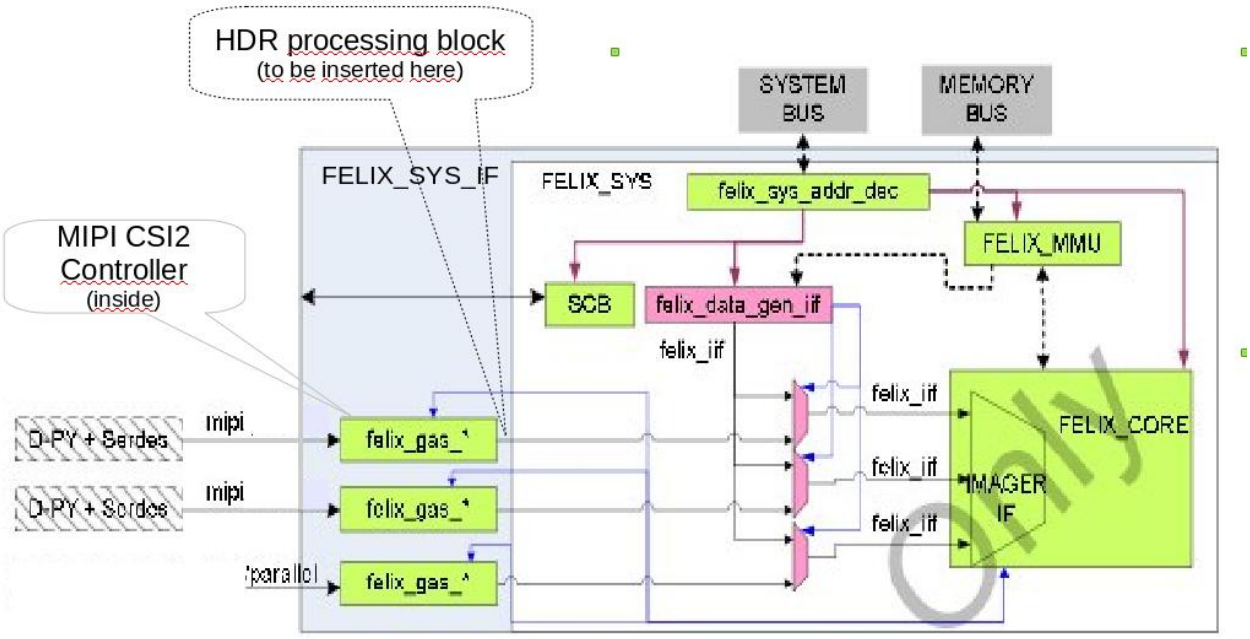


Рисунок 16.8. Интеграция блока MC_HDR в ISP

Данные с выхода блока MC_HDR поступают на вход felix_sys через bypass мультиплексор, управляемый битом HDR_ENABLE регистра HDR_CONTROL(0x008). После сброса бит HDR_ENABLE = 0, при этом данные с выхода felix_gasket_mipi поступают на вход felix_sys в обход блока MC_HDR. При установке бита HDR_ENABLE в 1 данные на вход felix_sys поступают с выхода блока MC_HDR.

16.7 Регистры

Перечень регистров приведен в Таблица 16.38.

Таблица 16.38. Перечень программно-доступных регистров

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
REG	Регистр	W/R	0x0	0x0

16.7.1 Регистр REG

Регистр REG.

Формат регистра REG приведен в Таблица 16.39.

Таблица 16.39. Формат регистра REG

Номер бита	Условное Обозначение	Назначение
0	FIELD	Бит
31:0	-	Резерв

17. КОНТРОЛЛЕР ДИСПЛЕЯ (DISPLAY)

17.1 Процессор дисплея

17.1.1 Общая информация

Обработка данных для вывода на дисплей:

- Поток видео UltraHD 4K 30 fps.
- Композиция до 7 уровней: видео1, видео2, графика и смарт-уровень до 4-х прямоугольников.
- Поддержка AFBC кодирования, масштабирования и вращения.
- Защита выводимой информации (protected display, trusted display).
- Параллельный интерфейс дисплея.

Структурная схема процессора дисплея представлена на Рисунок 17.1. Подробное описание блока содержится в документе «ARM® Mali™-DP550 Display Processor. Technical Reference Manual».

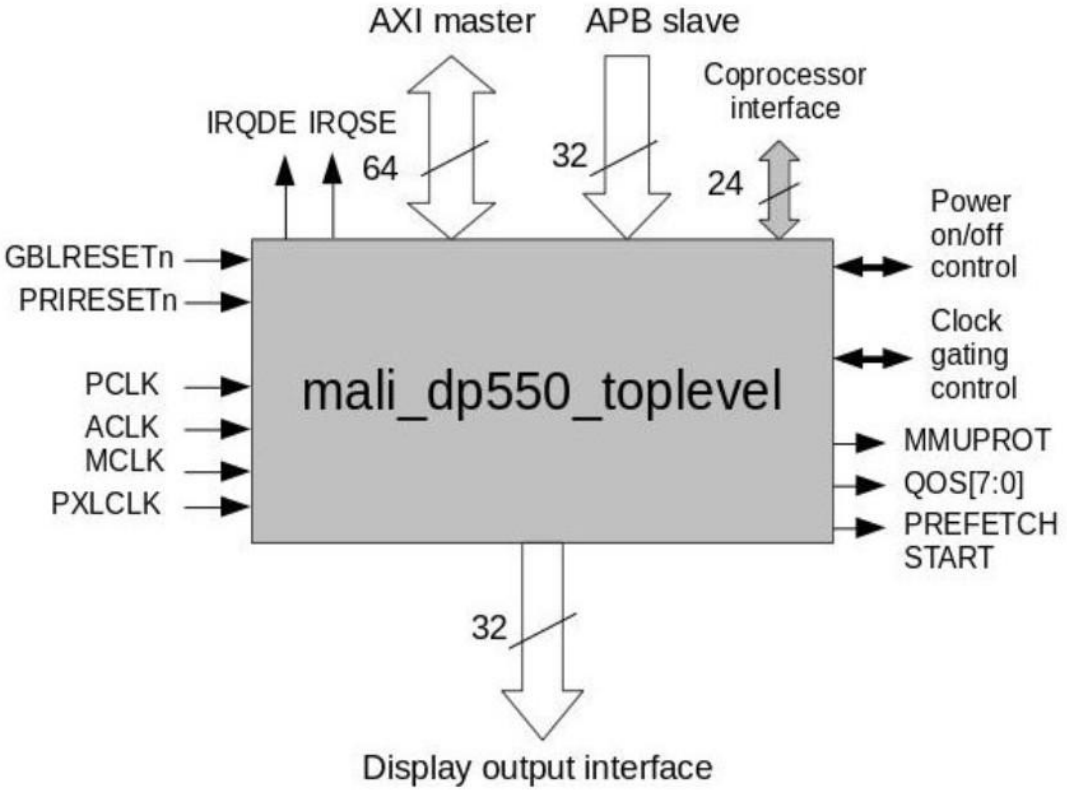


Рисунок 17.1. Структурная схема Mali-DP550

17.1.2 Конфигурационные параметры блока

Таблица 17.1. Конфигурационные параметры Mali-DP550

Наименование	Описание	Диапазон значений	Значение
CONFIG_SEC_DISPLAY_EN	Включает второе ядро	0, 1	0
CONFIG_MAX_LN_SIZE	Максимальная длина строки	1280, 2048, 4096	4096
CONFIG_IFIFO_SIZE	Количество слов во входном FIFO (latency buffer)	320, 512,1024	1024
CONFIG_AFBC_DBUFF_EN	Интеграция второго блока RAM для выходного буфера AFBC	0, 1 (для этого релиза 1)	1
Значеиия после сброса protected NSAID			
CONFIG_LV1_PROT_NSAID_PRI	Значение protected NSAID для слоя video1	0-15	0
CONFIG_LV2_PROT_NSAID_PRI	Значение protected NSAID для слоя video2	0-15	1
CONFIG_LG_PROT_NSAID_PRI	Значение protected NSAID для слоя graphics	0-15	2
CONFIG_LS_PROT_NSAID_PRI	Значение protected NSAID для слоя smart	0-15	3
CONFIG_MW_PROT_NSAID_PRI	Значение protected NSAID для слоя memory write	0-15	4
Значеиия после сброса protected StreamID			
CONFIG_LV1_PROT_STREAMID_PRI	Значение protected STREAMID для слоя video1	0-15	0
CONFIG_LV2_PROT_STREAMID _PRI	Значение protected STREAMID для слоя video2	0-15	1
CONFIG_LG_PROT_STREAMID _PRI	Значение protected STREAMID для слоя graphics	0-15	2
CONFIG_LS_PROT_STREAMID _PRI	Значение protected STREAMID для слоя smart	0-15	3
CONFIG_MW_PROT_STREAMID _PRI	Значение protected STREAMID для слоя memory write	0-15	4
Значеиия после сброса non- protected NSAID			
CONFIG_LV1_NPROT_NSAID_PRI	Значение non-protected NSAID для слоя video1	0-15	8
CONFIG_LV2_NPROT_NSAID_PRI	Значение non-protected NSAID для слоя video2	0-15	9
CONFIG_LG_NPROT_NSAID_PRI	Значение non-protected NSAID для слоя graphics	0-15	10
CONFIG_LS_NPROT_NSAID_PRI	Значение non-protected NSAID для слоя smart	0-15	11

Наименование	Описание	Диапазон значений	Значение
CONFIG_MW_NPROT_NSAID_PRI	Значение non-protected NSAID для слоя memory write	0-15	12
Значеиия после сброса non-protected StreamID			
CONFIG_LV1_NPROT_STREAMID_PRI	Значение non-protected STREAMID для слоя video1	0-15	8
CONFIG_LV2_NPROT_STREAMID _PRI	Значение non-protected STREAMID для слоя video2	0-15	9
CONFIG_LG_NPROT_STREAMID _PRI	Значение non-protected STREAMID для слоя graphics	0-15	10
CONFIG_LS_NPROT_STREAMID _PRI	Значение non-protected STREAMID для слоя smart	0-15	11
CONFIG_MW_NPROT_STREAMID _PRI	Значение non-protected STREAMID для слоя memory write	0-15	12

17.1.3 Интерфейсы блока

17.1.3.1 Глобальные сигналы

Таблица 17.2. Глобальные сигналы

Наименование	Направление	Описание
MCLK0	in	Главный синхросигнал обработки данных
PXLCLK0	in	Пиксельный синхросигнал
GBLRESETn	in	Глобальный асинхронный сброс, активный низким уровнем
PRIRESETn	in	Асинхронный сброс первичного домена питания

17.1.4 Сигналы прерывания

Таблица 17.3. Сигналы прерывания

Наименование	Направление	Описание
IRQDE0	out	Сигнал прерывания от display core
IRQSE0	out	Сигнал прерывания от скейлера

17.1.4.1 Сигналы AXI Master

17.1.4.2 Сигналы APB

17.1.4.3 Сигналы управления включением/выключением питания

Таблица 17.4. Сигналы управления включением/выключением питания

Наименование	Направление	Описание
GLBQREQn	in	Сигнал запроса выключения питания, активный низким уровнем
GLBQACCEPTn	out	Сигнал подтверждения запроса выключения питания, активный

Наименование	Направление	Описание
		низким уровнем
GLBQDENY	out	Отклонение запроса выключения питания
GLBQACTIVE	out	Индикатор активности глобального домена птиания

17.1.4.4 Сигналы управления включением/выключением синхросигнала ACLK0

Таблица 17.5. Сигналы управления включением/выключением синхросигнала ACLK0

Наименование	Направление	Описание
ACLK0QREQn	in	Сигнал запроса выключения ACLK0, активный низким уровнем
ACLK0QACCEPTn	out	Сигнал подтверждения запроса выключения ACLK0, активный низким уровнем
ACLK0QDENY	out	Отклонение запроса выключения ACLK0
ACLK0QACTIVE	out	Индикатор активности глобального домена ACLK0

17.1.4.5 Сигналы интерфейса с сопроцессором

Интерфейс с сопроцессором позволяет подключить блок для дополнительной обработки данных, такой как управление LCD backlight, tone remapping, подстройка контраста и и тд. Блок-схема интерфейса представлена на Рисунок 17.2.

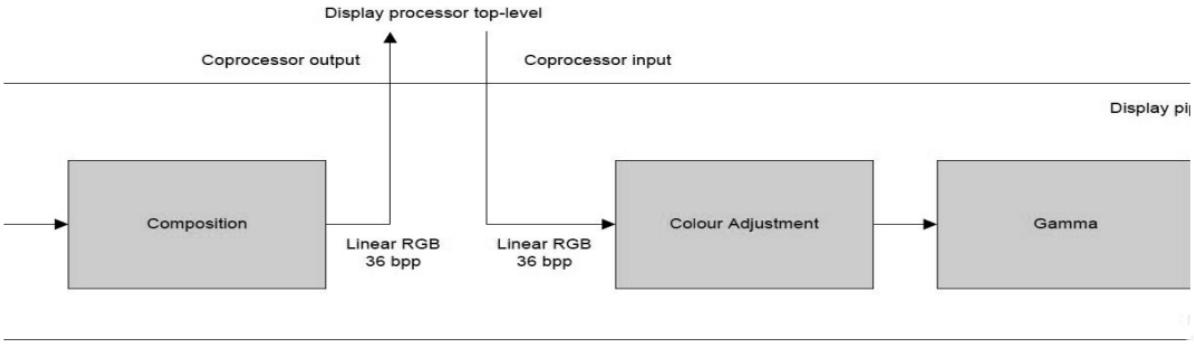


Рисунок 17.2. Блок-схема интерфейса с сопроцессором

Таблица 17.6. Сигналы интерфейса с сопроцессором

Наименование	Направление	Описание
CPIVALID0	in	Индикатор валидности данных от сопроцессора
CPIREADY0	out	Индикатор готовности контроллера дисплея
CPIEOF0	in	Индикатор конца кадра
CPIDATAR0[11:0]	in	Входные данные, красный компонент пикселя
CPIDATAG0[11:0]	in	Входные данные, зеленый компонент пикселя
CPIDATAB0[11:0]	in	Входные данные, синий компонент пикселя
CPOVALID0	out	Индикатор валидности данных от контроллера дисплея

Наименование	Направление	Описание
CPOREADY0	in	Индикатор готовности сопроцессора
CPOEOF0	out	Индикатор конца кадра
CPODATAR0[11:0]	out	Выходные данные, красный компонент пикселя
CPODATAG0[11:0]	out	Выходные данные, зеленый компонент пикселя
CPODATAB0[11:0]	out	Выходные данные, синий компонент пикселя

17.1.4.6 Сигналы интерфейса с дисплеем (совместим с MIPI DPI ?)

Таблица 17.7. Сигналы интерфейса с дисплеем

Наименование	Направление	Описание
DISPLREADY0	in	Флаг готовности дисплея
FRAMESTART0	out	Старт кадра
HSYNC0	out	Сигнал горизонтальной синхронизации
VSYNC0	out	Сигнал вертикальной синхронизации
DATAEN0	out	Сигнал разрешения передачи данных
PXLDATAR0[11:0]	out	Красный компонент пикселя
PXLDATAG0[11:0]	out	Зеленый компонент пикселя
PXLDATAB0[11:0]	out	Синий компонент пикселя
DISPUSER[3:0]	out	Сигналы общего назначения

17.1.4.7 Прочие сигналы

Таблица 17.8. Прочие сигналы

Наименование	Направление	Описание
MMUPROT0	out	Флаг MMU protection
QOS0[7:0]	out	Индикатор quality of service
PREFETCHSTART0	out	Флаг начала выборки данных

17.1.5 Частоты синхросигналов

Таблица 17.9. Частоты синхросигналов

Наименование синхросигнала	Частота (МГц)
ACLK0	400
PCLK	200
MCLK0	400
PXLCLK0	400

17.1.6 Регистры

Перечень регистров приведен в Таблица 17.10.

Таблица 17.10. Перечень программно-доступных регистров

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
REG	Регистр	W/R	0x0	0x0

17.1.6.1 Регистр REG

Регистр REG.

Формат регистра REG приведен в Таблица 17.11.

Таблица 17.11. Формат регистра REG

Номер бита	Условное обозначение	Назначение
0	FIELD	Бит
31:0	-	Резерв

17.2 Контроллер MIPI DSI

Основные особенности блока:

- Полная совместимость со стандартом MIPI Aliance Display Serial Interfase (DSI) Version 1.2.
- Поддержка всех команд, определенных в спецификации MIPI Aliance Display Command Set (DCS) Version 1.2:
  - передача всех команд по APB интерфейсу;
  - передача команд в low-power и high speed режимах во время передачи видео.
- Поддержка скорости передачи до 2500 Mbps на каждый lane.
- Поддержка до 4-х lanes.
- Двунаправленная передача и escape mode поддерживается для lane 0.
- Поддержка не-непрерывного DPHY синхросигнала для снижения энергопотребления.
- Поддержка режима Ulra Low Power.
- Вычисление ECC и контрольной суммы.
- Поддержка EndOfTransmission Packet.
- Поддержка 3D transmission.
- Генератор тестовых изображений.

Структурная схема DWC MIPI DSI представлена на Рисунок 17.3. Подробное описание блока содержится в документе «DesignWare Cores MIPI DSI Host Controller.Databook»

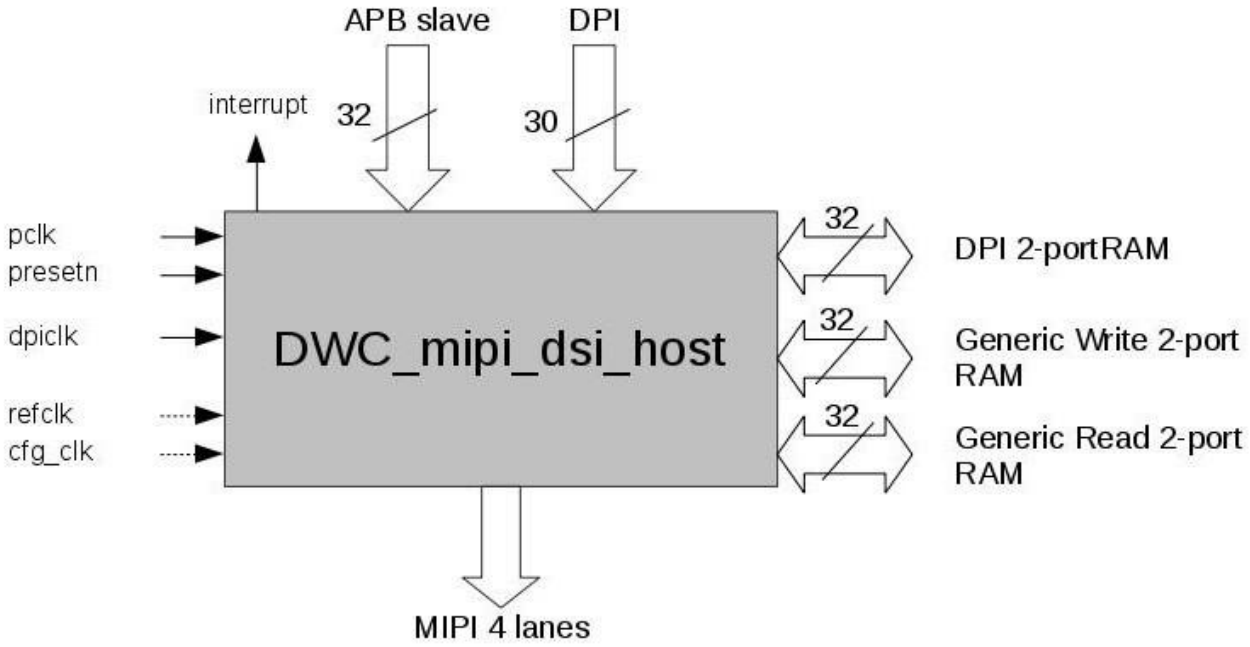


Рисунок 17.3. Структурная схема DWC MIPI DSI [1]host controller

17.2.1 Конфигурационные параметры блока

Таблица 17.12. Конфигурационные параметры DWC MIPI DSI host controller

Наименование	Описание	Диапазон значений	Значение
DSI_HOST_SNPS_PHY	Использовать PHY от Synopsys	0,1	1
DSI_HOST_DFLT_F_SYNC_TYPE	Количество стадий синхронизации между доменами	2-4	2
DSI_HOST_NUMBER_OF_LANES	Количество лэйнов	1-4	4
DSI_HOST_FPGA	Использовать дочернюю карту для FPGA прототипирования	0,1	0
DSI_HOST_DATAINTERFACE	Системный интерфейс данных	DPI, DBI, DBI&DPI, eDPI	DPI
DSI_HOST_SNPS_SYNC_RD_FIFOS	Тип памяти для FIFO	Асинхронный,синхронный	Синхронный
DSI_HOST_PIXELMEMADDRDEPTH	Глубина FIFO DPI данных	0 … 4096	3840 (2880)
DSI_HOST_GENERICCMDADDRDEPTH	Глубина Generic command FIFO	8 … 1024	10

Наименование	Описание	Диапазон значений	Значение
DSI_HOST_GENERICPLDADDRDEPTH	Глубина Generic Payload FIFO	8 … 4096	200
DSI_HOST_GENREADPLDADDRDEPTH	Глубина Generic Read FIFO	8 … 1024	16

17.2.2 Интерфейсы блока

17.2.2.1 Глобальные сигналы

Таблица 17.13. Глобальные сигналы

Наименование	Направление	Описание
Pclk	in	Синхросигнал шины APB
Presetn	in	Асинхронный сигнал сброса
Dpipclk	in	Синхросигнал DPI интерфейса
Refclk	in	Опорная частота для PLL
cfg_clk	in	Конфигурационный клок для DPHY

17.2.2.2 Сигналы прерывания

Таблица 17.14. Сигналы прерывания

Наименование	Направление	Описание
Interrupt	out	Запрос на прерывание

17.2.2.3 Сигналы APB

17.2.2.4 Сигналы DPI

Таблица 17.15. Сигналы DPI

Наименование	Направление	Описание
Dpivsync	in	Сигнал вертикальной синхронизации
Dpihsync	in	Сигнал горизонтальной синхронизации
dpipixdata[29:0]	in	Видеоданные
Dpidataen	in	Сигнал разрешения видеоданных
Dpishutdn	in	Инициирует Shutdown дисплея
Dpicolorm	in	Сигнал переключения между нормальным и reduced color режимами
Dpiupdatecfg	in	Сигнал указывает,что сделующий кадр будет с новой конфигурацией видео
Edpihalt	out	Индикатор остановки видеоинтерфейса

17.2.2.5 Сигналы интерфейса с 2-портовой памятью DPI данных

Таблица 17.16. Сигналы интерфейса с 2-портовой памятью DPI данных

Наименование	Направление	Описание
dpipixelwclk	out	Синхросигнал записи в память
dpipixelrclk	out	Синхросигнал чтения из памяти
dpipixelmemin[31:0]	in	Шина чтения памяти
dpipixelmemout[31:0]	out	Шина записи в память
dpipixelrdaddr[(DSI_HOST_PIXELMEMADDRWIDTH-1):0]	out	Адрес чтения из памяти
dpipixelwraddr[(DSI_HOSTPIXELMEMADDRWIDTH-1):0]	out	Адрес записи в память
dpipixelwrenable	out	Сигнал разрешения записи в память
dpipixelrdenable	out	Сигнал разрешения чтения из памяти
dpipixelmemselect	out	Сигнал выбора интерфейса с памятью

17.2.2.6 Сигналы интерфейса с 2-портовой памятью generic payload

Таблица 17.17. Сигналы интерфейса с 2-портовой памятью generic payload

Наименование	Направление	Описание
Dpipldwclk	out	Синхросигнал записи в память
Dpipldrclk	out	Синхросигнал чтения из памяти
dpipldmemin[31:0]	in	Шина чтения памяти
dpipldmemout[31:0]	out	Шина записи в память
dpipldrdaddr[(DSI_HOST_GENERICPLDMEMADDRWIDTH-1):0]	out	Адрес чтения из памяти
dpipldwraddr[(DSI_HOST_GENERICPLDMEMADDRWIDTH-1):0]	out	Адрес записи в память
Dpipldwrenable	out	Сигнал разрешения записи в память
Dpipldrdenable	out	Сигнал разрешения чтения из памяти
Dpipldmemselect	out	Сигнал выбора интерфейса с памятью

17.2.2.7 Сигналы интерфейса с 2-портовой памятью generic read

Таблица 17.18. Сигналы интерфейса с 2-портовой памятью generic read

Наименование	Направление	Описание
Dpireadwclk	out	Синхросигнал записи в память
Dpireadrclk	out	Синхросигнал чтения из памяти
dpireadmemin[31:0]	in	Шина чтения памяти
dpireadmemout[31:0]	out	Шина записи в память
dpireadrdaddr[(DSI_HOST_GENERICREADMEMADDRWIDTH-1):0]	out	Адрес чтения из памяти

Наименование	Направление	Описание
dpireadwraddr[(DSI_HOSTGENERICREADMEMADDRWIDTH-1):0]	out	Адрес записи в память
dpireadwrenable	out	Сигнал разрешения записи в память
dpireadrdenable	out	Сигнал разрешения чтения из памяти
dpireadmemselect	out	Сигнал выбора интерфейса с памятью

17.2.2.8 Сигналы PLL

Таблица 17.19. Сигналы PLL

Наименование	Направление	Описание
clkout_gp	out	Вспомогательный синхросигнал с частотой VCO/2
gp_clk_en	in	Разрешение синхросигнала clkout_gp
lock_pll	out	1 — PLL захватил фазу
n_obs[3:0]	out	Значение делителя частоты N (регистр PHY)
m_obs[9:0]	out	Значение умножителя частоты M (регистр PHY)
vco_cntrl_obs[5:0]	out	Операционный диапазон VCO (регистр PHY)
prop_cntrl_obs[5:0]	out	Пропорциональное управление charge pump (регистр PHY)
int_cntrl_obs[5:0]	out	Интегральное управление charge pump (регистр PHY)
gmp_cntrl_obs[1:0]	out	Эффективное сопротивление loop-фильтра = 1/gmp (регистр PHY)
cpbias_cntrl_out	out	Управление charge pump bias (регистр PHY)
cfgclkfreqrange[7:0]	in	Входная референсная частота round[(Fcfg_clk(MHz)-17)*4]
hsfreqrange[6:0]	in	Частота функционирования DPHY Для скорости 2500 Mbps = 0x49

17.2.2.9 Сигналы непрерывного тестирования

Таблица 17.20. Сигналы непрерывного тестирования

Наименование	Направление	Описание
cont_en	in	Сигнал разрешения непрерывного тестирования
cont_data[10:0]	out	Отображение в цифровом виде состояния аналоговых сигналов (rext, clkn, clkp, datan0-3, datap0-3)

17.2.2.10 Питание и аналоговые сигналы

Таблица 17.21. Питание и аналоговые сигналы

Наименование	Направление	Описание
vph	inout	Аналоговое высоковольтное питание (1.8 V)
vp	inout	Аналоговое низковольтное питание (0.9 V)
vpl	inout	Аналоговое низковольтное питание для PLL (0.9 V)
vdd	inout	Цифровое питание (0.9 V)
gd	inout	Аналоговая земля
vss	inout	Цифровая земля
atp	inout	Аналоговый тестовый вывод
rext	inout	Внешний референсный резистор

17.2.2.11 Сигналы MIPI 4 lanes

17.2.3 Частоты синхросигналов

Таблица 17.22. Частоты синхросигналов

Наименование синхросигнала	Частота (МГц)
pclk	200
dpiclk	297.4
refclk	24
cfg_clk	20

17.2.4 Регистры MIPI DSI

Этот раздел описывает все возможные регистры системы. Они организованы иерархически в карту и отдельные блоки.

Условия существования, упоминаемые в описаниях регистров и полей, указывают комбинации параметров конфигурации, необходимых для работы регистра или поля для существования в карте. Эти выражения валидны только в локальном контексте. Например, условия существования поля бита регистра записаны в предположении, что регистр существует и не включают условия существования самого регистра.

Карта регистров DSI приведена в Таблица 17.23.

Таблица 17.23. Карта регистров DSI

Регистр	Описание	Смещение	Тип доступа
VERSION	Содержит версию контроллера DSI.	0x0	R

PWR_UP	Управляет запуском ядра.	0x04	R/W
CLKMGR_CFG	Конфигурирует фактор деления синхросигнала lanebyteclk, который используется внутренними делителями частоты для организации задержек (timeout).	0x8	R/W
DPI_VCID	Конфигурирует идентификатор виртуального канала (Virtual Channel ID) для передачи данных от DPI.	0xC	R/W
DPI_COLOR_CODING	Конфигурирует цветовое кодирование DPI.	0x10	R/W
DPI_CFG_POL	Конфигурирует полярность сигналов DPI.	0x14	R/W
DPI_LP_CMD_TIM	Конфигурирует задержки для команд режима низкого питания, отсылаемых в видеорежиме.	0x18	R/W
DBI_VCID	Конфигурирует идентификатор виртуального канала (Virtual Channel ID) для передачи данных от DBI.	0x1c	R/W
DBI_CFG	Конфигурирует битовую ширину пикселей для DBI.	0x20	R/W
DBI_PARTITIONING_EN	Конфигурирует, следует ли DWC_mipi_dsi_host разделять DBI-траффик автоматически.	0x24	R/W
DBI_CMDSIZE	Конфигурирует размер команд и размеры частей, на которые следует автоматически разделять пакеты DBI.	0x28	R/W
PCKHDL_CFG	Конфигурирует то, как следует использовать сообщения EoTp, BTA, CRC и ECC для удовлетворения требованиям внешних устройств.	0x2c	R/W
GEN_VCID	Конфигурирует идентификатор виртуального канала для ответов на команду READ.	0x30	R/W
MODE_CFG	Конфигурирует режим работы: переключает между видеорежимом и командным режимом (отправка команд в видеорежиме тоже возможна).	0x34	R/W
VID_MODE_CFG	Конфигурирует несколько аспектов работы видеорежима	0x38	R/W
VID_PKT_SIZE	Конфигурирует размер видеопакета.	0x3c	R/W
VID_NUM_CHUNKS	Конфигурирует число используемых блоков.	0x40	R/W
VID_NULL_SIZE	Конфигурирует размер пустых пакетов.	0x44	R/W
VID_HSA_TIME	Конфигурирует период активной горизонтальной (строчной) синхронизации (HSA).	0x48	R/W
VID_HBP_TIME	Конфигурирует период горизонтальной (строчной) задней площадки гасящего импульса (HBP).	0x4c	R/W
VID_HLINE_TIME	Конфигурирует общую длительность каждой видеолинии.	0x50	R/W
VID_VSA_LINES	Конфигурирует период активной вертикальной (кадровой) синхронизации (VSA).	0x54	R/W
VID_VBP_LINES	Конфигурирует период вертикальной (кадровой) задней площадки гасящего импульса (VBP).	0x58	R/W
VID_VFP_LINES	Конфигурирует период вертикальной (кадровой) передней площадки гасящего импульса (VFP).	0x5c	R/W

VID_VACTIVE_LINES	Конфигурирует вертикальное разрешение видео.	0x60	R/W
EDPI_CMD_SIZE	Конфигурирует размер пакетов интерфейса eDPI.	0x64	R/W
CMD_MODE_CFG	Конфигурирует несколько аспектов работы командного режима.	0x68	R/W
GEN_HDR	Задает заголовок новых пакетов, отправленных через стандартный интерфейс.	0x6c	R/W
GEN_PLD_DATA	Задает полезную нагрузку для пакетов, отправленных через стандартный интерфейс.	0x70	R/W
CMD_PKT_STATUS	Содержит информацию о статусе буферов FIFO, связанных с интерфейсом DBI и со стандартным интерфейсом.	0x74	R
TO_CNT_CFG	Конфигурирует счетчики ошибок таймаута.	0x78	R/W
HS_RD_TO_CNT	Конфигурирует таймаут ответа внешних устройств после операций чтения в высокоскоростном режиме.	0x7c	R/W
LP_RD_TO_CNT	Конфигурирует таймаут ответа внешних устройств после операций чтения в режиме низкого питания.	0x80	R/W
HS_WR_TO_CNT	Конфигурирует таймаут ответа внешних устройств после операций записи в высокоскоростном режиме.	0x84	R/W
LP_WR_TO_CNT	Конфигурирует таймаут ответа внешних устройств после операций записи в режиме низкого питания.	0x88	R/W
BTA_TO_CNT	Конфигурирует таймаут ответа внешних устройств после завершения процесса обхода шины (Bus Turnaround).	0x8c	R/W
SDF_3D	Хранит данные настроек 3D для пакетов VSS в видеорежиме.	0x90	R/W
LPCLK_CTRL	Конфигурирует возможность использования непостоянного синхросигнала в тракте синхросигналов.	0x94	R/W
PHY_TMR_LPCLK_CFG	Задает время, которым DWC_mipi_dsi_host руководствуется в расчетах для переключения тракта синхросигналов между высокоскоростным режимом и режимом низкого питания.	0x98	R/W
PHY_TMR_CFG	Задает время, которым DWC_mipi_dsi_host руководствуется в расчетах для переключения трактов данных между высокоскоростным режимом и режимом низкого питания.	0x9c	R/W
PHY_RSTZ	Управляет сбросами и PLL (ФАПЧ - фазовой автоподстройкой частоты) для D-PHY.	0xa0	R/W
PHY_IF_CFG	Конфигурирует количество активных трактов и минимальное время пребывания в состоянии остановки.	0xa4	R/W
PHY_ULPS_CTRL	Конфигурирует вход и выход из ультра-низкого режима питания (ULPS) в D-PHY.	0xa8	R/W
PHY_TX_TRIGGERS	Конфигурирует контакты, активирующие триггеры в D-PHY.	0xac	R/W
PHY_STATUS	Содержит информацию о состояниях D-PHY.	0xb0	R
PHY_TST_CTRL0	Управляет контактами синхросигналов и контактами сброса, относящимися к интерфейсу D-PHY, зависящему от поставщика.	0xb4	R/W

PHY_TST_CTRL1	Управляет контактами данных и разрешающими контактами, относящимися к интерфейсу D-PHY, зависящему от поставщика.	0xb8	R/W
INT_ST0	Содержит статусы источников прерываний из отчетов о подтверждениях и из D-PHY.	0xbc	R
INT_ST1	Содержит статусы источников прерываний, связанных с таймутами, ECC, CRC, размерами пакетов, EoTp, стандартным и DBI-интерфейсами.	0xc0	R
INT_MSK0	Конфигурирует маски для источников прерываний, которые затрагивают регистр INT_ST0	0xc4	R/W
INT_MSK1	Конфигурирует маски для источников прерываний, которые затрагивают регистр INT_ST1.	0xc8	R/W
PHY_CAL	Управляет калибровкой фазового сдвига D-PHY.	0xcc	R/W
INT_FORCE0	Назначает прерывания, которые затрагивают регистр INT_ST0.	0xd8	W
INT_FORCE1	Назначает прерывания, которые затрагивают регистр INT_ST1.	0xdc	W
DSC_PARAMETER	Конфигурирует сжатие потока диспеля (DSC).	0xf0	R/W
PHY_TMR_RD_CFG	Конфигурирует временные интервалы, относящиеся к PHY, для выполнения некоторых операций в тактах байтового синхросигнала тракта.	0xf4	R/W
VID_SHADOW_CTRL	Управляет функциями дублирования интерфейса DPI.	0x100	R/W
DPI_VCID_ACT	Хранит значение, которое контроллер использует для DPI_VCID.	0x10c	R
DPI_COLOR_CODING_ACT	Хранит значение, которое контроллер использует для DPI_COLOR_CODING.	0x110	R
DPI_LP_CMD_TIM_ACT	Хранит значение, которое контроллер использует для DPI_LP_CMD_TIM.	0x118	R
VID_MODE_CFG_ACT	Хранит значение, которое контроллер использует для VID_MODE_CFG.	0x138	R
VID_PKT_SIZE_ACT	Хранит значение, которое контроллер использует для VID_PKT_SIZE.	0x13c	R
VID_NUM_CHUNKS_ACT	Хранит значение, которое контроллер использует для VID_NUM_CHUNKS.	0x140	R
VID_NULL_SIZE_ACT	Хранит значение, которое контроллер использует для VID_NULL_SIZE.	0x144	R
VID_HSA_TIME_ACT	Хранит значение, которое контроллер использует для VID_HSA_TIME.	0x148	R
VID_HBP_TIME_ACT	Хранит значение, которое контроллер использует для VID_HBP_TIME.	0x14c	R
VID_HLINE_TIME_ACT	Хранит значение, которое контроллер использует для VID_HLINE_TIME.	0x150	R
VID_VSA_LINES_ACT	Хранит значение, которое контроллер использует для VID_VSA_LINES.	0x154	R
VID_VBP_LINES_ACT	Хранит значение, которое контроллер использует для VID_VBP_LINES.	0x158	R

VID_VFP_LINES_ACT	Хранит значение, которое контроллер использует для VID_VFP_LINES.	0x15c	R
VID_VACTIVE_LINES_ACT	Хранит значение, которое контроллер использует для VID_VACTIVE_LINES.	0x160	R
SDF_3D_ACT	Хранит значение, которое контроллер использует для SDF_3D.	0x190	R

17.2.4.1 Регистр VERSION

Этот регистр содержит версию контроллера DSI.

Поля регистра VERSION описаны в Таблица 17.24.

Таблица 17.24. Регистр VERSION

Диапазон	Название	Описание	Тип доступа	Исходное значение
[31:0]	version	Это поле указывает версию DWC_mipi_dsi_host.	R	0x3133312a

17.2.4.2 Регистр PWR_UP

Этот регистр управляет запуском ядра.

Поля регистра PWR_UP описаны в Таблица 17.25.

Таблица 17.25. Регистр PWR_UP

Диапазон	Название	Описание	Тип доступа	Исходное значение
[0]	shutdownz	Этот бит конфигурирует ядро либо для запуска, либо для сброса. 0 - сброс; 1 - запуск.	R/W	0
[31:1]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.3 Регистр CLKMGR_CFG

Этот регистр конфигурирует фактор деления синхросигнала lanebyteclk, который используется внутренними делителями частоты для организации задержек (timeout).

Поля регистра CLKMGR_CFG описаны в Таблица 17.26.

Таблица 17.26. Регистр CLKMGR_CFG

Диапазон	Название	Описание	Тип доступа	Исходное значение
[7:0]	tx_esc_clk_division	Это поле указывает фактор деления для источника снихронизирующих импульсов TX Escape (lanebyteclk). Значения "0" и "1" в этом поле прекращают генерацию импульсов TX_ESC.	R/W	0

Диапазон	Название	Описание	Тип доступа	Исходное значение
[15:8]	to_clk_division	Это поле указывает фактор деления для синхросигнала Time Out. Этот сигнал используется для синхронизации задержек при конфигурировании ошибки перехода из высокоскоростного (HS) режима в режим низкого питания (LP) и из режима низкого питания (LP) в высокоскоростной (HS).	R/W	0
[31:16]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.4 Регистр DPI_VCID

Этот регистр конфигурирует идентификатор виртуального канала (Virtual Channel ID) для передачи данных от DPI.

Условия существования: ((DSI_HOST_DATAINTERFACE == 2) || (DSI_HOST_DATAINTERFACE == 3) || (DSI_HOST_DATAINTERFACE == 4))==1.

Поля регистра DPI_VCID описаны в Таблица 17.27.

Таблица 17.27. Регистр DPI_VCID

Диапазон	Название	Описание	Тип доступа	Исходное значение
[1:0]	dpi_vcid	Это поле конфигурирует идентификатор виртуального канала DPI, который индексируется к пакетам видеорежима.	R/W	0
[31:2]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.5 Регистр DPI_COLOR_CODING

Этот регистр конфигурирует цветовое кодирование DPI.

Условия существования: ((DSI_HOST_DATAINTERFACE == 2) || (DSI_HOST_DATAINTERFACE == 3) || (DSI_HOST_DATAINTERFACE == 4))==1.

Поля регистра DPI_COLOR_CODING описаны в Таблица 17.28.

Таблица 17.28. Регистр DPI_COLOR_CODING

Диапазон	Название	Описание	Тип доступа	Исходное значение
[3:0]	dpi_color_coding	Это поле конфигурирует цветовое кодирование DPI следующим образом: 0000: 16-битное, конфигурация 1; 0001: 16-битное, конфигурация 2; 0010: 16-битное, конфигурация 3; 0011: 18-битное, конфигурация 1; 0100: 18-битное, конфигурация 2; 0101: 24-битное; 0110: 20-битное, кодирование YCbCr 4:2:2, свободно упакованное (loosely packed); 0111: 24-битное YCbCr 4:2:2; 1000: 16-битное YCbCr 4:2:2; 1001: 30-битное; 1010: 36-битное; 1011: 12-битное YCbCr 4:2:0; 11xx: сжатые данные DSC24. Если выбран интерфейс eDPI и работа ведется в командном режиме (cmd_video_mode = 1), то: 0000: 16-битное, конфигурация 1; 0001: 16-битное, конфигурация 2; 0010: 16-битное, конфигурация 3; 0011: 18-битное, конфигурация 1; 0100: 18-битное, конфигурация 2; 0101-0111: 24-битное; 1Xxx: сжатые данные DSC24.	R/W	0
[7:4]	-	Зарезервировано для дальнейшего использования.		0
[8]	loosely18_en	Когда установлено значение "1", этот бит активирует свободно упакованный вариант для 18-битных конфигураций.	R/W	0
[31:9]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.6 Регистр DPI_CFG_POL

Этот регистр конфигурирует полярность сигналов DPI.

Условия существования: ((DSI_HOST_DATAINTERFACE == 2) || (DSI_HOST_DATAINTERFACE == 3) || (DSI_HOST_DATAINTERFACE == 4))==1.

Поля регистра DPI_CFG_POL описаны в Таблица 17.29.

Таблица 17.29. Регистр DPI_CFG_POL

Диапазон	Название	Описание	Тип доступа	Исходное значение
[0]	dataen_active_low	Когда установлено значение "1", этот бит конфигурирует активный низкий уровень на контакте подключения данных (dpidataen).	R/W	0
[1]	vsync_active_low	Когда установлено значение "1", этот бит конфигурирует активный низкий уровень на контакте вертикальной (кадровой) синхронизации (dpivsync).	R/W	0

Диапазон	Название	Описание	Тип доступа	Исходное значение
[2]	hsync_active_low	Когда установлено значение "1", этот бит конфигурирует активный низкий уровень на контакте горизонтальной (строчной) синхронизации (dpihsync).	R/W	0
[3]	shutd_active_low	Когда установлено значение "1", этот бит конфигурирует активный низкий уровень на контакте отключения питания (dpishutdn).	R/W	0
[4]	colorm_active_low	Когда установлено значение "1", этот бит конфигурирует активный низкий уровень на контакте цветового режима (dpicolorm).	R/W	0
[31:5]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.7 Регистр DPI_LP_CMD_TIM

Конфигурирует задержки для команд режима низкого питания, отсылаемых в видеорежиме.

Условия существования: ((DSI_HOST_DATAINTERFACE == 2) || (DSI_HOST_DATAINTERFACE == 3) || (DSI_HOST_DATAINTERFACE == 4))==1.

Поля регистра DPI_LP_CMD_TIM описаны в Таблица 17.30.

Таблица 17.30. Регистр DPI_LP_CMD_TIM

Диапазон	Название	Описание	Тип доступа	Исходное значение
[7:0]	invact_lpcmd_time	Это поле используется для передачи команд в режиме низкого питания. Оно определяет в байтах размер наибольшего пакета, который может быть помещен на линию во время VACT (Vertical Active)-зоны.	R/W	0
[15:8]	-	Зарезервировано для дальнейшего использования.		0
[23:16]	outvact_lpcmd_time	Это поле используется для передачи команд в режиме низкого питания. Оно определяет в байтах размер наибольшего пакета, который может быть помещен на линию во время VSA, VBP, и VFP-зон.	R/W	0
[31:24]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.8 Регистр DBI_VCID

Этот регистр конфигурирует идентификатор виртуального канала (Virtual Channel ID) для передачи данных от DBI (display bus interface).

Условия существования: ((DSI_HOST_DATAINTERFACE == 1) || (DSI_HOST_DATAINTERFACE == 3))==1.

Поля регистра DBI_VCID описаны в Таблица 17.31.

Таблица 17.31. Регистр DBI_VCID

Диапазон	Название	Описание	Тип доступа	Исходное значение
[1:0]	dbi_vcid	Это поле конфигурирует идентификатор виртуального канала, который индексируется к пакетам DCS (display command set) от интерфейса DBI.	R/W	0
[31:2]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.9 Регистр DBI_CFG

Этот регистр конфигурирует битовую ширину пикселей для DBI.

Условия существования: ((DSI_HOST_DATAINTERFACE == 1) || (DSI_HOST_DATAINTERFACE == 3))==1.

Поля регистра DBI_CFG описаны в Таблица 17.32.

Таблица 17.32. Регистр DBI_CFG

Диапазон	Название	Описание	Тип доступа	Исходное значение
[3:0]	in_dbi_conf	Это поле конфигурирует входные данные по пикселям DBI следующим образом: 0000: 8-бит, 8 бит на пиксель (bpp); 0001: 8-бит, 12 бит на пиксель ; 0010: 8-бит, 16 бит на пиксель; 0011: 8-бит, 18 бит на пиксель; 0100: 8-бит, 24 бита на пиксель; 0101: 9-бит, 18 бит на пиксель; 0110: 16-бит, 8 бит на пиксель; 0111: 16-бит, 12 бит на пиксель; 1000: 16-бит, 16 бит на пиксель; 1001: 16-бит, 18 бит на пиксель, опция 1; 1010: 16-бит, 18 бит на пиксель, опция 2; 1011: 16-бит, 24 бита на пиксель, опция 1; 1100: 16-бит, 24 бита на пиксель, опция 2.	R/W	0
[7:4]	-	Зарезервировано для дальнейшего использования.		0
[11:8]	out_dbi_conf	Это поле конфигурирует выходные данные по пикселям DBI следующим образом: 0000: 8-бит, 8 бит на пиксель (bpp); 0001: 8-бит, 12 бит на пиксель ; 0010: 8-бит, 16 бит на пиксель; 0011: 8-бит, 18 бит на пиксель; 0100: 8-бит, 24 бита на пиксель; 0101: 9-бит, 18 бит на пиксель; 0110: 16-бит, 8 бит на пиксель; 0111: 16-бит, 12 бит на пиксель; 1000: 16-бит, 16 бит на пиксель; 1001: 16-бит, 18 бит на пиксель, опция 1; 1010: 16-бит, 18 бит на пиксель, опция 2; 1011: 16-бит, 24 бита на пиксель, опция 1; 1100: 16-бит, 24 бита на пиксель, опция 2.	R/W	0
[15:12]	-	Зарезервировано для дальнейшего использования.		0

Диапазон	Название	Описание	Тип доступа	Исходное значение
[17:16]	lut_size_conf	Это поле конфигурирует размер, используемый для транспортировки команд записи в таблицы просмотра (Write Lut commands) следующим образом: 00: 16-битное отображение цветов; 01: 18-битное отображение цветов; 10: 24-битное отображение цветов; 11: 16-битное отображение цветов.	R/W	0
[31:18]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.10 Регистр DBI_PARTITIONING_EN

Этот регистр конфигурирует, следует ли DWC_mipi_dsi_host разделять DBI-траффик автоматически.

Условия существования: ((DSI_HOST_DATAINTERFACE == 1) || (DSI_HOST_DATAINTERFACE == 3))==1.

Поля регистра DBI_PARTITIONING_EN описаны в Таблица 17.33.

Таблица 17.33. Регистр DBI_PARTITIONING_EN

Диапазон	Название	Описание	Тип доступа	Исходное значение
[0]	partitioning_en	Когда установлено значение "1", этот бит подключает использование команд ввода write_memory_continue (системе необходимо обеспечить правильное разделение команд длинной записи (Long Write commands)). Когда бит не установлен, DWC_mipi_dsi_host производит разделение автоматически.	R/W	0
[31:1]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.11 Регистр DBI_CMDSIZE

Этот регистр конфигурирует размер команд и размеры частей, на которые следует автоматически разделять пакеты DBI.

Условия существования: ((DSI_HOST_DATAINTERFACE == 1) || (DSI_HOST_DATAINTERFACE == 3))==1.

Поля регистра DBI_CMDSIZE описаны в Таблица 17.34.

Таблица 17.34. Регистр DBI_CMDSIZE

Диапазон	Название	Описание	Тип доступа	Исходное значение
[15:0]	wr_cmd_size	Это поле конфигурирует размер команд DCS для записи в память. Размер полезной нагрузки пакета DSI - это фактический размер полезной нагрузки минус 1, потому что команда DCS включена в полезную нагрузку пакета DSI.	R/W	0

Диапазон	Название	Описание	Тип доступа	Исходное значение
[31:16]	allowed_cmd_size	Это поле конфигурирует максимальный разрешенный размер команды DCS для записи в память. Это поле используется для разделения команды записи в память на одну команду write_memory_start и переменное число команд write_memory_continue. Оно используется только если бит partitioning_en регистра DBI_CFG отключен. Размер полезной нагрузки пакета DSI - это фактический размер полезной нагрузки минус 1, потому что команда DCS включена в полезную нагрузку пакета DSI.	R/W	0

17.2.4.12 Регистр PCKHDL_CFG

Этот регистр конфигурирует то, как следует использовать сообщения EoTp, BTA, CRC и ECC для удовлетворения требованиям внешних устройств.

Поля регистра PCKHDL_CFG описаны в Таблица 17.35.

Таблица 17.35. Регистр PCKHDL_CFG

Диапазон	Название	Описание	Тип доступа	Исходное значение
[0]	eotp_tx_en	Когда установлено значение 1, этот бит разрешает передачу пакета окончания передачи EoTp (End of Transmission packet).	R/W	0
[1]	eotp_rx_en	Когда установлено значение 1, этот бит разрешает получение пакета окончания передачи EoTp.	R/W	0
[2]	bta_en	Когда установлено значение 1, этот бит разрешает запрос обхода шины BTA (Bus Turn-Around).	R/W	0
[3]	ecc_rx_en	Когда установлено значение 1, этот бит разрешает получение кода коррекции ошибок ECC (Error-Correcting Code), коррекцию ошибки и отчёт о ней.	R/W	0
[4]	crc_rx_en	Когда установлено значение 1, этот бит разрешает получение ошибки CRC (циклического избыточного кода) и отправку отчёта об ошибке.	R/W	0
[31:5]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.13 Регистр GEN_VCID

Этот регистр конфигурирует идентификатор виртуального канала для ответов на команду READ, которые следует сохранять и возвращать в стандартный (Generic) интерфейс.

Поля регистра GEN_VCID описаны в Таблица 17.36.

Таблица 17.36. Регистр GEN_VCID

Диапазон	Название	Описание	Тип доступа	Исходное значение
[1:0]	gen_vcid_rx	Это поле указывает на идентификатор виртуального канала в процессе обратного чтения в стандартном интерфейсе.	R/W	0
[31:2]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.14 Регистр MODE_CFG

Этот регистр конфигурирует режим работы: переключает между видеорежимом и командным режимом (отправка команд в видеорежиме тоже возможна).

Поля регистра MODE_CFG описаны в Таблица 17.37.

Таблица 17.37. Регистр MODE_CFG

Диапазон	Название	Описание	Тип доступа	Исходное значение
[0]	cmd_video_mode	Этот бит конфигурирует режим работы: 0: видеорежим; 1: командный режим.	R/W	0x1
[31:1]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.15 Регистр VID_MODE_CFG

Этот регистр конфигурирует несколько аспектов работы видеорежима: режим передачи, переключение на низкое питание в середине кадра, разрешение подтверждений и отправка команд при низком питании.

Условия существования: ((DSI_HOST_DATAINTERFACE == 2) || (DSI_HOST_DATAINTERFACE == 3) || (DSI_HOST_DATAINTERFACE == 4))==1.

Поля регистра VID_MODE_CFG описаны в Таблица 17.38.

Таблица 17.38. Регистр VID_MODE_CFG

Диапазон	Название	Описание	Тип доступа	Исходное значение
[1:0]	vid_mode_type	Это поле задает тип передачи при видеорежиме следующим образом: 00: немонопольный с синхронизирующими импульсами; 01: немонопольный с синхронизирующими событиями; 10 и 11: пакетный (монопольный) режим.	R/W	0
[7:2]	-	Зарезервировано для дальнейшего использования.		0
[8]	lp_vsa_en	Когда установлено значение "1", этот бит разрешает возвращение к режиму низкого питания в рамках периода VSA (vertical sync active), при условии, что временные периоды это позволяют.	R/W	0
[9]	lp_vbp_en	Когда установлено значение "1", этот бит разрешает возвращение к режиму низкого питания в рамках периода VBP (vertical back porch), при условии, что временные периоды это позволяют.	R/W	0
[10]	lp_vfp_en	Когда установлено значение "1", этот бит разрешает возвращение к режиму низкого питания в рамках периода VFP (vertical front porch), при условии, что временные периоды это позволяют.	R/W	0

Диапазон	Название	Описание	Тип доступа	Исходное значение
[11]	lp_vact_en	Когда установлено значение "1", этот бит разрешает возвращение к режиму низкого питания в рамках периода VACT (vertical active), при условии, что временные периоды это позволяют.	R/W	0
[12]	lp_hbp_en	Когда установлено значение "1", этот бит разрешает возвращение к режиму низкого питания в рамках периода HBP (horizontal back porch), при условии, что временные периоды это позволяют.	R/W	0
[13]	lp_hfp_en	Когда установлено значение "1", этот бит разрешает возвращение к режиму низкого питания в рамках периода HFP (horizontal front porch), при условии, что временные периодыи это позволяют.	R/W	0
[14]	frame_bta_ack_en	Когда установлено значение "1", этот бит разрешает запрос на подтверждающий ответ в конце кадра.	R/W	0
[15]	lp_cmd_en	Когда установлено значение "1", этот бит разрешает передачу команд только в режиме низкого питания.	R/W	0
[16]	vpg_en	Когда установлено значение "1", этот бит разрешает генератор паттернов в видеорежиме.	R/W	0
[19:17]	-	Зарезервировано для дальнейшего использования.		0
[20]	vpg_mode	Это поле используется для выбора паттерна изображения: 0: цветовые полосы (горизонтальный или вертикальный); 1: паттерн BER (коэффициента битовых ошибок) (только вертикальный).	R/W	0
[23:21]	-	Зарезервировано для дальнейшего использования.		0
[24]	vpg_orientation	Это поле задает ориентацию цветовых полос следующим образом: 0: вертикальный режим; 1: горизонтальный режим.	R/W	0
[31:25]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.16 Регистр VID_PKT_SIZE

Этот регистр конфигурирует размер видеопакета.

Условия существования: ((DSI_HOST_DATAINTERFACE == 2) || (DSI_HOST_DATAINTERFACE == 3) || (DSI_HOST_DATAINTERFACE == 4))==1.

Поля регистра VID_PKT_SIZE описаны в Таблица 17.39.

Таблица 17.39. Регистр VID_PKT_SIZE

Диапазон	Название	Описание	Тип доступа	Исходное значение
[13:0]	vid_pkt_size	Это поле конфигурирует число пикселей в одном видеопакете. - для 18-битных, не свободно упакованных типов данных, это число должно быть кратно 4; - для типов данных YCbCr, это число должно быть кратно двум, в соответствии с рекомендациями в спецификации DSI; - для DSC24 это поле конфигурирует количество байтов. Следует отметить, что при использовании срезов (также называемых блоками) размер минимального пакета должен быть не меньше 3.	R/W	0
[31:14]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.17 Регистр VID_NUM_CHUNKS

Этот регистр конфигурирует число используемых блоков. Размер данных в каждом блоке задан в VID_PKT_SIZE.

Условия существования: ((DSI_HOST_DATAINTERFACE == 2) || (DSI_HOST_DATAINTERFACE == 3) || (DSI_HOST_DATAINTERFACE == 4))==1.

Поля регистра VID_NUM_CHUNKS описаны в Таблица 17.40.

Таблица 17.40. Регистр VID_NUM_CHUNKS

Диапазон	Название	Описание	Тип доступа	Исходное значение
[12:0]	vid_num_chunks	Этот регистр конфигурирует число блоков, которые необходимо передать в течение периода Line (блок - это пара из видеопакета и пустого пакета). Если установлены значения "0" или "1", видеолиния по-прежнему передается одним пакетом. Если установлено значение "1", такой пакет является частью блока. Это значит, что за ним следует пустой пакет (при условии, что vid_null_size>0). В остальных случаях для передачи каждой видеолинии используется набор блоков.	R/W	0
[31:13]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.18 Регистр VID_NULL_SIZE

Этот регистр конфигурирует размер пустых пакетов.

Условия существования: ((DSI_HOST_DATAINTERFACE == 2) || (DSI_HOST_DATAINTERFACE == 3) || (DSI_HOST_DATAINTERFACE == 4))==1.

Поля регистра VID_NULL_SIZE описаны в Таблица 17.41.

Таблица 17.41. Регистр VID_NULL_SIZE

Диапазон	Название	Описание	Тип доступа	Исходное значение
[12:0]	vid_null_size	Этот регистр конфигурирует число байтов внутри пустого пакета. Установка значения "0" отключает пустые пакеты.	R/W	0
[31:13]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.19 Регистр VID_HSA_TIME

Этот регистр конфигурирует период активной горизонтальной (строчной) синхронизации (HSA).

Условия существования: ((DSI_HOST_DATAINTERFACE == 2) || (DSI_HOST_DATAINTERFACE == 3) || (DSI_HOST_DATAINTERFACE == 4))==1.

Поля регистра VID_HSA_TIME описаны в Таблица 17.42.

Таблица 17.42. Регистр VID_HSA_TIME

Диапазон	Название	Описание	Тип доступа	Исходное значение
[11:0]	vid_hsa_time	Это поле конфигурирует период активной горизонтальной (строчной) синхронизации (HSA) в тактах байтового синхросигнала тракта.	R/W	0
[31:12]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.20 Регистр VID_HBP_TIME

Этот регистр конфигурирует период горизонтальной (строчной) задней площадки гасящего импульса (HBP).

Условия существования: ((DSI_HOST_DATAINTERFACE == 2) || (DSI_HOST_DATAINTERFACE == 3) || (DSI_HOST_DATAINTERFACE == 4))==1.

Поля регистра VID_HBP_TIME описаны в Таблица 17.43.

Таблица 17.43. Регистр VID_HBP_TIME

Диапазон	Название	Описание	Тип доступа	Исходное значение
[11:0]	vid_hbp_time	Это поле конфигурирует период горизонтальной (строчной) задней площадки гасящего импульса (HBP) в тактах байтового синхросигнала тракта.	R/W	0
[31:12]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.21 Регистр VID_HLINE_TIME

Этот регистр конфигурирует общую длительность каждой видеолинии.

Условия существования: ((DSI_HOST_DATAINTERFACE == 2) || (DSI_HOST_DATAINTERFACE == 3) || (DSI_HOST_DATAINTERFACE == 4))==1.

Поля регистра VID_HLINE_TIME описаны в Таблица 17.44.

Таблица 17.44. Регистр VID_HLINE_TIME

Диапазон	Название	Описание	Тип доступа	Исходное значение
[14:0]	vid_hline_time	Это поле конфигурирует длительность общего периода видеолинии (HSA+HBP+HACT+HFP) в тактах байтового синхросигнала тракта.	R/W	0
[31:15]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.22 Регистр VID_VSA_LINES

Этот регистр конфигурирует период активной вертикальной (кадровой) синхронизации (VSA).

Условия существования: ((DSI_HOST_DATAINTERFACE == 2) || (DSI_HOST_DATAINTERFACE == 3) || (DSI_HOST_DATAINTERFACE == 4))==1.

Поля регистра VID_VSA_LINES описаны в Таблица 17.45.

Таблица 17.45. Регистр VID_VSA_LINES

Диапазон	Название	Описание	Тип доступа	Исходное значение
[9:0]	vsa_lines	Это поле конфигурирует период активной вертикальной (кадровой) синхронизации (VSA), измеряемый числом горизонтальных линий.	R/W	0
[31:10]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.23 Регистр VID_VBP_LINES

Этот регистр конфигурирует период вертикальной (кадровой) задней площадки гасящего импульса (VBP).

Условия существования: ((DSI_HOST_DATAINTERFACE == 2) || (DSI_HOST_DATAINTERFACE == 3) || (DSI_HOST_DATAINTERFACE == 4))==1.

Поля регистра VID_VBP_LINES описаны в Таблица 17.46.

Таблица 17.46. Регистр VID_VBP_LINES

Диапазон	Название	Описание	Тип доступа	Исходное значение
[9:0]	vbp_lines	Это поле конфигурирует период вертикальной (кадровой) задней площадки гасящего импульса (VBP), измеряемый числом горизонтальных линий.	R/W	0
[31:10]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.24 Регистр VID_VFP_LINES

Этот регистр конфигурирует период вертикальной (кадровой) передней площадки гасящего импульса (VFP).

Условия существования: ((DSI_HOST_DATAINTERFACE == 2) || (DSI_HOST_DATAINTERFACE == 3) || (DSI_HOST_DATAINTERFACE == 4))==1.

Поля регистра VID_VFP_LINES описаны в Таблица 17.47.

Таблица 17.47. Регистр VID_VFP_LINES

Диапазон	Название	Описание	Тип доступа	Исходное значение
[9:0]	vfp_lines	Это поле конфигурирует период вертикальной (кадровой) передней площадки гасящего импульса (VFP), измеряемый числом горизонтальных линий.	R/W	0
[31:10]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.25 Регистр VID_VACTIVE_LINES

Этот регистр конфигурирует вертикальное разрешение видео.

Условия существования: ((DSI_HOST_DATAINTERFACE == 2) || (DSI_HOST_DATAINTERFACE == 3) || (DSI_HOST_DATAINTERFACE == 4))==1.

Поля регистра VID_VACTIVE_LINES описаны в Таблица 17.48.

Таблица 17.48. Регистр VID_VACTIVE_LINES

Диапазон	Название	Описание	Тип доступа	Исходное значение
[13:0]	v_active_lines	Это поле конфигурирует период вертикальной активности (Vertical Active), измеряемый числом горизонтальных линий.	R/W	0
[31:14]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.26 Регистр EDPI_CMD_SIZE

Этот регистр конфигурирует размер пакетов интерфейса eDPI.

Условия существования: ((DSI_HOST_DATAINTERFACE == 4))==1.

Поля регистра EDPI_CMD_SIZE описаны в Таблица 17.49.

Таблица 17.49. Регистр EDPI_CMD_SIZE

Диапазон	Название	Описание	Тип доступа	Исходное значение
[15:0]	edpi_allowed_cmd_size	Это поле задает максимальный разрешенный размер команды записи в память от интерфейса eDPI, измеряемый в пикселях. Автоматическое разделение данных, полученных от eDPI, подключено перманентно.	R/W	0
[31:16]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.27 Регистр CMD_MODE_CFG

Этот регистр конфигурирует несколько аспектов работы командного режима: эффект разрыва (tearing effect), подтверждение каждого пакета и скоростной режим для передачи каждого типа данных, относящегося к командам.

Поля регистра CMD_MODE_CFG описаны в Таблица 17.50.

Таблица 17.50. Регистр CMD_MODE_CFG

Диапазон	Название	Описание	Тип доступа	Исходное значение
[0]	tear_fx_en	Когда установлено значение "1", этот бит разрешает запрос на подтверждение эффекта разрыва.	R/W	0
[1]	ack_rqst_en	Когда установлено значение "1", этот бит разрешает запрос на подтверждение после каждой передачи пакета.	R/W	0
[7:2]	-	Зарезервировано для дальнейшего использования.		0
[8]	gen_sw_0p_tx	Этот бит конфигурирует тип передачи команды стандартного пакета короткой записи, не содержащий параметров. 0: высокоскоростной режим; 1: режим низкого питания.	R/W	0
[9]	gen_sw_1p_tx	Этот бит конфигурирует тип передачи команды стандартного пакета короткой записи, содержащий один параметр. 0: высокоскоростной режим; 1: режим низкого питания.	R/W	0
[10]	gen_sw_2p_tx	Этот бит конфигурирует тип передачи команды стандартного пакета короткой записи, содержащий два параметра. 0: высокоскоростной режим; 1: режим низкого питания.	R/W	0
[11]	gen_sr_0p_tx	Этот бит конфигурирует тип передачи команды стандартного пакета короткого чтения, не содержащий параметров. 0: высокоскоростной режим; 1: режим низкого питания.	R/W	0
[12]	gen_sr_1p_tx	Этот бит конфигурирует тип передачи команды стандартного пакета короткого чтения, содержащий один параметр. 0: высокоскоростной режим; 1: режим низкого питания.	R/W	0
[13]	gen_sr_2p_tx	Этот бит конфигурирует тип передачи команды стандартного пакета короткого чтения, содержащий два параметра. 0: высокоскоростной режим; 1: режим низкого питания.	R/W	0
[14]	gen_lw_tx	Этот бит конфигурирует тип передачи команды стандартного пакета длинной записи. 0: высокоскоростной режим; 1: режим низкого питания.	R/W	0
[15]	-	Зарезервировано для дальнейшего использования.		0
[16]	dcs_sw_0p_tx	Этот бит конфигурирует тип передачи команды пакета короткой записи от DCS, не содержащий параметров. 0: высокоскоростной режим; 1: режим низкого питания.	R/W	0

Диапазон	Название	Описание	Тип доступа	Исходное значение
[17]	dcs_sw_1p_tx	Этот бит конфигурирует тип передачи команды пакета короткой записи от DCS, содержащий один параметр. 0: высокоскоростной режим; 1: режим низкого питания.	R/W	0
[18]	dcs_sr_0p_tx	Этот бит конфигурирует тип передачи команды пакета короткого чтения от DCS, не содержащий параметров. 0: высокоскоростной режим; 1: режим низкого питания.	R/W	0
[19]	dcs_lw_tx	Этот бит конфигурирует тип передачи команды пакета длинной записи от DCS. 0: высокоскоростной режим; 1: режим низкого питания.	R/W	0
[23:20]	-	Зарезервировано для дальнейшего использования.		0
[24]	max_rd_pkt_size	Этот бит конфигурирует тип передачи команды чтения пакета максимального размера. 0: высокоскоростной режим; 1: режим низкого питания.	R/W	0
[31:25]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.28 Регистр GEN_HDR

Этот регистр устанавливает заголовок для новых пакетов, отправленных через стандартный интерфейс.

Поля регистра GEN_HDR описаны в Таблица 17.51.

Таблица 17.51. Регистр GEN_HDR

Диапазон	Название	Описание	Тип доступа	Исходное значение
[5:0]	gen_dt	Это поле конфигурирует тип данных пакета заголовка.	R/W	0
[7:6]	gen_vc	Это поле конфигурирует идентификатор виртуального канала пакета заголовка.	R/W	0
[15:8]	gen_wc_lsbyte	Это поле конфигурирует: 1) для длинных пакетов: младший байт данных счетчика слов заголовка пакета; 2) для коротких пакетов: байт data 0.	R/W	0
[23:16]	gen_wc_msbyte	Это поле конфигурирует: 1) для длинных пакетов: старший байт данных счетчика слов заголовка пакета; 2) для коротких пакетов: байт data 1.	R/W	0
[31:24]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.29 Регистр GEN_PLD_DATA

Этот регистр задает полезную нагрузку для пакетов, отправленных через стандартный интерфейс. При чтении возвращает содержимое ответов на команду READ от внешних устройств.

Поля регистра GEN_PLD_DATA описаны в Таблица 17.52.

Таблица 17.52. Регистр GEN_PLD_DATA

Диапазон	Название	Описание	Тип доступа	Исходное значение
[7:0]	gen_pld_b1	Это поле указывает на 1-й байт полезной нагрузки пакета.	R/W	0
[15:8]	gen_pld_b2	Это поле указывает на 2-й байт полезной нагрузки пакета.	R/W	0
[23:16]	gen_pld_b3	Это поле указывает на 3-й байт полезной нагрузки пакета.	R/W	0
[31:24]	gen_pld_b4	Это поле указывает на 4-й байт полезной нагрузки пакета.	R/W	0

17.2.4.30 Регистр CMD_PKT_STATUS

Этот регистр содержит информацию о статусе буферов FIFO, связанных с интерфейсом DBI и со стандартным интерфейсом.

Поля регистра CMD_PKT_STATUS описаны в Таблица 17.53.

Таблица 17.53. Регистр CMD_PKT_STATUS

Диапазон	Название	Описание	Тип доступа	Исходное значение
[0]	gen_cmd_empty	Этот бит указывает, что стандартная команда FIFO пуста.	R	0x1
[1]	gen_cmd_full	Этот бит указывает, что стандартная команда FIFO полна.	R	0
[2]	gen_pld_w_empty	Этот бит указывает, что стандартная полезная нагрузка буфера FIFO при записи пуста.	R	0x1
[3]	gen_pld_w_full	Этот бит указывает, что стандартная полезная нагрузка буфера FIFO при записи полна.	R	0
[4]	gen_pld_r_empty	Этот бит указывает, что стандартная полезная нагрузка буфера FIFO при чтении пуста.	R	0x1
[5]	gen_pld_r_full	Этот бит указывает, что стандартная полезная нагрузка буфера FIFO при чтении полна.	R	0
[6]	gen_rd_cmd_busy	Этот бит устанавливается, когда происходит отправка команды чтения и её очистка в момент сохранения полного ответа в буфере FIFO стандартного интерфейса.	R	0
[7]	-	Зарезервировано для дальнейшего использования.		0
[8]	dbi_cmd_empty	Этот бит указывает, что команда FIFO интерфейса DBI пуста. Условия существования: ((DSI_HOST_DATAINTERFACE == 1) || (DSI_HOST_DATAINTERFACE == 3))==1.	R	0x1
[9]	dbi_cmd_full	Этот бит указывает, что команда FIFO интерфейса DBI полна. Условия существования: ((DSI_HOST_DATAINTERFACE == 1) || (DSI_HOST_DATAINTERFACE == 3))==1.	R	0
[10]	dbi_pld_w_empty	Этот бит указывает, что полезная нагрузка буфера FIFO интерфейса DBI при записи пуста. Условия существования: ((DSI_HOST_DATAINTERFACE == 1) || (DSI_HOST_DATAINTERFACE == 3))==1.	R	0x1

Диапазон	Название	Описание	Тип доступа	Исходное значение
[11]	dbi_pld_w_full	Этот бит указывает, что полезная нагрузка буфера FIFO интерфейса DBI при записи полна. Условия существования: ((DSI_HOST_DATAINTERFACE == 1) || (DSI_HOST_DATAINTERFACE == 3))==1.	R	0
[12]	dbi_pld_r_empty	Этот бит указывает, что полезная нагрузка буфера FIFO интерфейса DBI при чтении пуста. Условия существования: ((DSI_HOST_DATAINTERFACE == 1) || (DSI_HOST_DATAINTERFACE == 3))==1.	R	0x1
[13]	dbi_pld_r_full	Этот бит указывает, что полезная нагрузка буфера FIFO интерфейса DBI при чтении полна. Условия существования: ((DSI_HOST_DATAINTERFACE == 1) || (DSI_HOST_DATAINTERFACE == 3))==1.	R	0
[14]	dbi_rd_cmd_busy	Этот бит устанавливается, когда происходит отправка команды чтения и её очистка в момент сохранения полного ответа в буфере FIFO интерфейса DBI. Условия существования: ((DSI_HOST_DATAINTERFACE == 1) || (DSI_HOST_DATAINTERFACE == 3))==1.	R	0
[31:15]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.31 Регистр TO_CNT_CFG

Этот регистр конфигурирует счетчики ошибок таймаута. Они используются для предупреждения системы о сбоях путем прерывания, а в случае неожиданных ситуаций, которые провоцируют тупиковые условия, перезагружают ядро.

Поля регистра TO_CNT_CFG описаны в Таблица 17.54.

Таблица 17.54. Регистр TO_CNT_CFG

Диапазон	Название	Описание	Тип доступа	Исходное значение
[15:0]	lprx_to_cnt	Это поле конфигурирует счетчик таймаута, который запускает обнаружение конфликта таймаута при приеме в режиме низкого питания (измеряется в тактах TO_CLK_DIVISION).	R/W	0
[31:16]	hstx_to_cnt	Это поле конфигурирует счетчик таймаута, который запускает обнаружение конфликта таймаута при передаче в высокоскоростном режиме (измеряется в тактах TO_CLK_DIVISION). В немонопольном режиме: если времени недостаточно для переключения из режима HS в LP и обратно в течение периода, длящегося от окончания одной линии данных до начала синхронизации следующей, то DSI-соединение будет возвращать режим низкого питания один раз за кадр. Тогда следует сконфигурировать TO_CLK_DIVISION и hstx_to_cnt так, чтобы удовлетворять следующей формуле: hstx_to_cnt * lanebyteclkperiod * TO_CLK_DIVISION >= время передачи данных одного кадра * (1 + 10%).	R/W	0

Диапазон	Название	Описание	Тип доступа	Исходное значение
		В монопольном режиме: пакеты RGB-пикселей сжаты во времени, что оставляет больше времени на линию сканирования. Поэтому, если режим монопольный и времени достаточно, чтобы переключиться из режима HS в LP и обратно в течение периода, длящегося от окончания одной линии данных до начала синхронизации следующей, DSI-соединение может вернуться в режим LP и обратно за этот временной интервал для сохранения энергии. Если выбран этот вариант, следуте сконфигурировать TO_CLK_DIVISION и hstx_to_cnt так, чтобы удовлетворять следующей формуле: hstx_to_cnt * lanebyteclkperiod * TO_CLK_DIVISION >= время передачи данных одной линии * (1 + 10%).		

17.2.4.32 Регистр HS_RD_TO_CNT

Этот регистр конфигурирует таймаут ответа внешних устройств после операций чтения в высокоскоростном режиме.

Поля регистра HS_RD_TO_CNT описаны в Таблица 17.55.

Таблица 17.55. Регистр HS_RD_TO_CNT

Диапазон	Название	Описание	Тип доступа	Исходное значение
[15:0]	hs_rd_to_cnt	Это поле задает период, в течение которого DWC_mipi_dsi_host удерживает соединение после отправки команды чтения в высокоскоростном режиме. Этот период измеряется в тактах lanebyteclk, и начинает отсчитываться с момента, когда D-PHY входит в состояние остановки и не производит прерываний.	R/W	0
[31:16]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.33 Регистр LP_RD_TO_CNT

Этот регистр конфигурирует таймаут ответа внешних устройств после операций чтения в режиме низкого питания.

Поля регистра LP_RD_TO_CNT описаны в Таблица 17.56.

Таблица 17.56. Регистр LP_RD_TO_CNT

Диапазон	Название	Описание	Тип доступа	Исходное значение
[15:0]	lp_rd_to_cnt	Это поле задает период, в течение которого DWC_mipi_dsi_host удерживает соединение после отправки команды чтения в режиме низкого питания. Этот период измеряется в тактах lanebyteclk, и начинает отсчитываться с момента, когда D-PHY входит в состояние остановки и не производит прерываний.	R/W	0

Диапазон	Название	Описание	Тип доступа	Исходное значение
[31:16]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.34 Регистр HS_WR_TO_CNT

Этот регистр конфигурирует таймаут ответа внешних устройств после операций записи в высокоскоростном режиме.

Поля регистра HS_WR_TO_CNT описаны в Таблица 17.57.

Таблица 17.57. Регистр HS_WR_TO_CNT

Диапазон	Название	Описание	Тип доступа	Исходное значение
[15:0]	hs_wr_to_cnt	Это поле задает период, в течение которого DWC_mipi_dsi_host удерживает соединение после отправки команды записи в высокоскоростном режиме. Этот период измеряется в тактах lanebyteclk, и начинает отсчитываться с момента, когда D-PHY входит в состояние остановки и не производит прерываний.	R/W	0
[23:16]	-	Зарезервировано для дальнейшего использования.		0
[24]	presp_to_mode	Когда установлено значение "1", этот бит устраивает так, чтобы таймаут ответа от внешних устройств, спровоцированный hs_wr_to_cnt, использовался только один раз за кадр интерфейса eDPI, после выполнения следующих условий: - значение dpivsync_edpiwms выросло и упало; - пакеты от eDPI переданы и буфер FIFO интерфейса eDPI снова пуст. В этом сценарии к D-PHY не отправляются никакие запросы от интерфейсов помимо eDPI, даже если присутствуют данные, готовые к отправке от стандартного интерфейса или интерфейса DBI, что заставляет D-PHY вернуться в состояние остановки. Когда она возвращается в него, активируется счетчик PRESP_TO, и только после окончания его работы контроллер отсылает любые другие готовые данные. Условия существования: ((DSI_HOST_DATAINTERFACE == 4))==1.	R/W	0
[31:25]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.35 Регистр LP_WR_TO_CNT

Этот регистр конфигурирует таймаут ответа внешних устройств после операций записи в режиме низкого питания.

Поля регистра LP_WR_TO_CNT описаны в Таблица 17.58.

Таблица 17.58. Регистр LP_WR_TO_CNT

Диапазон	Название	Описание	Тип доступа	Исходное значение
[15:0]	lp_wr_to_cnt	Это поле задает период, в течение которого DWC_mipi_dsi_host удерживает соединение после отправки команды записи в режиме низкого питания. Этот период измеряется в тактах lanebyteclk, и начинает отсчитываться с момента, когда D-PHY входит в состояние остановки и не производит прерываний.	R/W	0
[31:16]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.36 Регистр BTA_TO_CNT

Этот регистр конфигурирует таймаут ответа внешних устройств после завершения процесса обхода шины (Bus Turnaround).

Поля регистра BTA_TO_CNT описаны в Таблица 17.59.

Таблица 17.59. Регистр BTA_TO_CNT

Диапазон	Название	Описание	Тип доступа	Исходное значение
[15:0]	bta_to_cnt	Это поле задает период, в течение которого DWC_mipi_dsi_host удерживает соединение после завершения процесса обхода шины. Этот период измеряется в тактах lanebyteclk, и начинает отсчитываться с момента, когда D-PHY входит в состояние остановки и не производит прерываний.	R/W	0
[31:16]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.37 Регистр SDF_3D

Этот регистр хранит данные настроек 3D для пакетов VSS (пакетов старта вертикальной синхронизации) в видеорежиме.

Условия существования: ((DSI_HOST_DATAINTERFACE == 2) || (DSI_HOST_DATAINTERFACE == 3) || (DSI_HOST_DATAINTERFACE == 4))==1.

Поля регистра SDF_3D описаны в Таблица 17.60.

Таблица 17.60. Регистр SDF_3D

Диапазон	Название	Описание	Тип доступа	Исходное значение
[1:0]	mode_3d	Это поле определяет, включен или отключен режим 3D и задает ориентацию дисплея. 00: 3D режим отключен (2D режим включен); 01: 3D режим включен, портретная ориентация; 10: 3D режим включен, альбомная ориентация; 11: зарезервировано.	R/W	0

Диапазон	Название	Описание	Тип доступа	Исходное значение
[3:2]	format_3d	Это поле задает формат 3D изображения: 00: линия (чередование линий левого и правого потока данных); 01: кадр (чередование кадров левого и правого потока данных). 10: пиксель (чередование пикселей левого и правого потока данных). 11: зарезервировано.	R/W	0
[4]	second_vsync	Это поле определяет присутствие второго сигнала VSYNC между левым и правым изображениями, когда выбран формат 3D-изображения на основе кадра. 0: между левым и правым потоком данных нет синхронизирующих импульсов; 1: между левым и правым потоком данных подключен синхросигнал (HSYNC, VSYNC, импульс гашения).	R/W	0
[5]	right_first	Этот бит задает порядок подачи левого/правого изображения: 0: первым подается изображение с левого глаза, затем - с правого; 1: первым подается изображение с правого глаза, затем - с левого.	R/W	0
[15:6]	-	Зарезервировано для дальнейшего использования.		0
[16]	send_3d_cfg	Когда установлен, устраивает так, чтобы последующий VSS-пакет включил полезную нагрузку 3D-настроек в каждый VSS-пакет.	R/W	0
[31:17]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.38 Регистр LPCLK_CTRL

Этот регистр конфигурирует возможность использования непостоянного синхросигнала в тракте синхросигналов.

Поля регистра LPCLK_CTRL описаны в Таблица 17.61.

Таблица 17.61. Регистр LPCLK_CTRL

Диапазон	Название	Описание	Тип доступа	Исходное значение
[0]	phy_txrequestclkhs	Этот бит управляет сигналом PPI (протокола интерфейса PHY) txrequestclkhs в D-PHY.	R/W	0
[1]	auto_clklane_ctrl	Этот бит подключает автоматический механизм прекращения подачи синхросигнала в тракт синхросигналов, когда временные рамки это позволяют.	R/W	0
[31:2]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.39 Регистр PHY_TMR_LPCLK_CFG

Этот регистр задает время, которым DWC_mipi_dsi_host руководствуется в расчетах для переключения тракта синхросигналов между высокоскоростным режимом и режимом низкого питания.

Условия существования: ((DSI_HOST_DATAINTERFACE == 2) || (DSI_HOST_DATAINTERFACE == 3) || (DSI_HOST_DATAINTERFACE == 4))==1.

Поля регистра PHY_TMR_LPCLK_CFG описаны в Таблица 17.62.

Таблица 17.62. Регистр PHY_TMR_LPCLK_CFG

Диапазон	Название	Описание	Тип доступа	Исходное значение
[9:0]	phy_clklp2hs_time	Это поле конфигурирует максимальное время, которое требуется для того, чтобы тракт синхросигналов D-PHY перешел из режима передачи данных при низком питании в высокоскоростной режим передачи. Оно измеряется в тактах байтового синхросигнала тракта.	R/W	0
[15:10]	-	Зарезервировано для дальнейшего использования.		0
[25:16]	phy_clkhs2lp_time	Это поле конфигурирует максимальное время, которое требуется для того, чтобы тракт синхросигналов D-PHY перешел из высокоскоростного режима передачи в режим передачи данных при низком питании. Оно измеряется в тактах байтового синхросигнала тракта.	R/W	0
[31:26]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.40 Регистр PHY_TMR_CFG

Этот регистр задает время, которым DWC_mipi_dsi_host руководствуется в расчетах для переключения трактов данных между высокоскоростным режимом и режимом низкого питания.

Условия существования: ((DSI_HOST_DATAINTERFACE == 2) || (DSI_HOST_DATAINTERFACE == 3) || (DSI_HOST_DATAINTERFACE == 4))==1.

Поля регистра PHY_TMR_CFG описаны в Таблица 17.63.

Таблица 17.63. Регистр PHY_TMR_CFG

Диапазон	Название	Описание	Тип доступа	Исходное значение
[9:0]	phy_lp2hs_time	Это поле конфигурирует максимальное время, которое требуется для того, чтобы тракты данных D-PHY перешли из режима передачи данных при низком питании в высокоскоростной режим передачи. Оно измеряется в тактах байтового синхросигнала тракта.	R/W	0
[15:10]	-	Зарезервировано для дальнейшего использования.		0
[25:16]	phy_hs2lp_time	Это поле конфигурирует максимальное время, которое требуется для того, чтобы тракты данных D-PHY перешли из высокоскоростного режима передачи в режим передачи данных при низком питании. Оно измеряется в тактах байтового синхросигнала тракта.	R/W	0
[31:26]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.41 Регистр PHY_RSTZ

Этот регистр управляет сбросами и PLL (ФАПЧ - фазовой автоподстройкой частоты) для D-PHY.

Поля регистра PHY_RSTZ описаны в Таблица 17.64.

Таблица 17.64. Регистр PHY_RSTZ

Диапазон	Название	Описание	Тип доступа	Исходное значение
[0]	phy_shutdownz	Когда установлено значение "0", этот бит отключает заполненный D-PHY Macro.	R/W	0
[1]	phy_rstz	Когда установлено значение "0", этот бит помещает в состояние сброса цифровой раздел D-PHY.	R/W	0
[2]	phy_enableclk	Когда установлено значение "1", этот бит подключает модуль синхросигнала тракта D-PHY (Clock Lane Module).	R/W	0
[3]	phy_forcepll	Когда D-PHY находится в режиме ULPS (режим ультра-низкого питания), этот бит подключает PLL D-PHY. Условия существования: DSI_HOST_FPGA==0.	R/W	0
[31:4]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.42 Регистр PHY_IF_CFG

Этот регистр конфигурирует количество активных трактов и минимальное время пребывания в состоянии остановки.

Поля регистра PHY_IF_CFG описаны в Таблица 17.65.

Таблица 17.65. Регистр PHY_IF_CFG

Диапазон	Название	Описание	Тип доступа	Исходное значение
[1:0]	n_lanes	Это поле конфигурирует количество активных трактов данных: 00: один тракт данных (тракт 0); 01: два тракта данных (тракты 0 и 1); 10: три тракта данных (тракты 0, 1, и 2); 11: четыре тракта данных (тракты 0, 1, 2 и 3).	R/W	DSI_HOST_NUMBER_OF_LANES-1
[7:2]	-	Зарезервировано для дальнейшего использования.		0
[15:8]	phy_stop_wait_time	Это поле конфигурирует минимальное время ожидания перед запросом высокоскоростной передачи данных после состояния остановки.	R/W	0
[31:16]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.43 Регистр PHY_ULPS_CTRL

Этот регистр конфигурирует вход и выход из режима ультранизкого питания (ULPS) в D-PHY.

Поля регистра PHY_ULPS_CTRL описаны в Таблица 17.66.

Таблица 17.66. Регистр PHY_ULPS_CTRL

Диапазон	Название	Описание	Тип доступа	Исходное значение
[0]	phy_txrequlpsclk	Запрос режима ULPS на тракте синхросигнала.	R/W	0
[1]	phy_txexitulpsclk	Выход из режима ULPS на тракте синхросигнала.	R/W	0
[2]	phy_txrequlpslan	Запрос режима ULPS на всех активных трактах данных.	R/W	0
[3]	phy_txexitulpslan	Выход из режима ULPS на всех активных трактах данных.	R/W	0
[31:4]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.44 Регистр PHY_TX_TRIGGERS

Этот регистр конфигурирует контакты, активирующие триггеры в D-PHY.

Поля регистра PHY_TX_TRIGGERS описаны в Таблица 17.67.

Таблица 17.67. Регистр PHY_TX_TRIGGERS

Диапазон	Название	Описание	Тип доступа	Исходное значение
[3:0]	phy_tx_triggers	Это поле управляет триггерными передачами.	R/W	0
[31:4]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.45 Регистр PHY_STATUS

Этот регистр содержит информацию о состояниях D-PHY.

Поля регистра PHY_STATUS описаны в Таблица 17.68.

Таблица 17.68. Регистр PHY_STATUS

Диапазон	Название	Описание	Тип доступа	Исходное значение
[0]	phy_lock	Этот бит указывает состояние сигнала phylock D-PHY.	R	0
[1]	phy_direction	Этот бит указывает состояние сигнала phydirection D-PHY.	R	0
[2]	phy_stopstateclklane	Этот бит указывает состояние сигнала phystopstateclklane D-PHY.	R	0
[3]	phy_ulpsactivenotclk	Этот бит указывает состояние сигнала phyulpsactivenotclk D-PHY.	R	0
[4]	phy_stopstate0lane	Этот бит указывает состояние сигнала phystopstate0lane D-PHY.	R	0
[5]	phy_ulpsactivenot0lane	Этот бит указывает состояние сигнала ulpsactivenot0lane D-PHY.	R	0
[6]	phy_rxulpsesc0lane	Этот бит указывает состояние сигнала rxulpsesc0lane D-PHY.	R	0x1
[7]	phy_stopstate1lane	Этот бит указывает состояние сигнала phystopstate1lane D-PHY. Условия существования: DSI_HOST_NUMBER_OF_LANES>1.	R	0

Диапазон	Название	Описание	Тип доступа	Исходное значение
[8]	phy_ulpsactivenot1lane	Этот бит указывает состояние сигнала ulpsactivenot1lane D-PHY. Условия существования: DSI_HOST_NUMBER_OF_LANES>1.	R	0x1
[9]	phy_stopstate2lane	Этот бит указывает состояние сигнала phystopstate2lane D-PHY. Условия существования: DSI_HOST_NUMBER_OF_LANES>2.	R	0
[10]	phy_ulpsactivenot2lane	Этот бит указывает состояние сигнала ulpsactivenot2lane D-PHY. Условия существования: DSI_HOST_NUMBER_OF_LANES>2.	R	0x1
[11]	phy_stopstate3lane	Этот бит указывает состояние сигнала phystopstate3lane D-PHY. Условия существования: DSI_HOST_NUMBER_OF_LANES>3.	R	0
[12]	phy_ulpsactivenot3lane	Этот бит указывает состояние сигнала ulpsactivenot3lane D-PHY. Условия существования: DSI_HOST_NUMBER_OF_LANES>3.	R	0x1
[31:13]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.46 Регистр PHY_TST_CTRL0

Этот регистр управляет контактами синхросигналов и контактами сброса, относящимися к интерфейсу D-PHY, зависящему от поставщика.

Поля регистра PHY_TST_CTRL0 описаны в Таблица 17.69.

Таблица 17.69. Регистр PHY_TST_CTRL0

Диапазон	Название	Описание	Тип доступа	Исходное значение
[0]	phy_testclr	Очистка тестового интерфейса PHY (сигнал с активным верхним уровнем).	R/W	0x1
[1]	phy_testclk	Этот бит используется для синхронизации шины TESTDIN с D-PHY.	R/W	0
[31:2]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.47 Регистр PHY_TST_CTRL1

Этот регистр управляет контактами данных и разрешающими контактами, относящимися к интерфейсу D-PHY, зависящему от поставщика.

Поля регистра PHY_TST_CTRL1 описаны в Таблица 17.70.

Таблица 17.70. Регистр PHY_TST_CTRL1

Диапазон	Название	Описание	Тип доступа	Исходное значение
[7:0]	phy_testdin	8-битная шина данных ввода тестового интерфейса PHY для внутреннего программирования регистров и доступа к тестовому функционалу.	R/W	0

Диапазон	Название	Описание	Тип доступа	Исходное значение
[15:8]	pht_testdout	8-битная шина данных вывода PHY для обратного считывания и функционала внутреннего тестирования.	R	0
[16]	phy_testen	Селектор команд тестового интерфейса PHY: 1: операция записи адреса устанавливается на заднем фронте сигнала testclk; 0: операция записи данных устанавливается на переднем фронте сигнала testclk.	R/W	0
[31:17]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.48 Регистр INT_ST0

Этот регистр содержит статусы источников прерываний из отчетов о подтверждениях и из D-PHY.

Поля регистра INT_ST0 описаны в Таблица 17.71.

Таблица 17.71. Регистр INT_ST0

Диапазон	Название	Описание	Тип доступа	Исходное значение
[0]	ack_with_err_0	Этот бит извлекает ошибку SoT из отчета об ошибке подтверждения.	R	0
[1]	ack_with_err_1	Этот бит извлекает ошибку синхронизации SoT Sync из отчета об ошибке подтверждения.	R	0
[2]	ack_with_err_2	Этот бит извлекает ошибку синхронизации EoT Sync из отчета об ошибке подтверждения.	R	0
[3]	ack_with_err_3	Этот бит извлекает ошибку команды входа в режим выхода (Escape Mode Entry Command) из отчета об ошибке подтверждения.	R	0
[4]	ack_with_err_4	Этот бит извлекает ошибку синхронизации при передаче в режиме низкого питания (LP Transmit Sync) из отчета об ошибке подтверждения.	R	0
[5]	ack_with_err_5	Этот бит извлекает ошибку таймаута внешних устройств (Peripheral Timeout) из отчета об ошибке подтверждения.	R	0
[6]	ack_with_err_6	Этот бит извлекает ошибку False Control из отчета об ошибке подтверждения.	R	0
[7]	ack_with_err_7	Этот бит извлекает зарезервированные данные (которые зависят от прибора) из отчета об ошибке подтверждения.	R	0
[8]	ack_with_err_8	Этот бит извлекает ошибку ECC, однобитную (обнаруженную и скорректированную) из отчета об ошибке подтверждения.	R	0
[9]	ack_with_err_9	Этот бит извлекает ошибку ECC, многобитную (обнаруженную, но не скорректированную) из отчета об ошибке подтверждения.	R	0
[10]	ack_with_err_10	Этот бит извлекает ошибку контрольной суммы (только длинный пакет) из отчета об ошибке подтверждения.	R	0
[11]	ack_with_err_11	Этот бит извлекает нераспознанный тип данных DSI из отчета об ошибке подтверждения.	R	0
[12]	ack_with_err_12	Этот бит извлекает невалидный идентификатор DSI VC из отчета об ошибке подтверждения.	R	0

Диапазон	Название	Описание	Тип доступа	Исходное значение
[13]	ack_with_err_13	Этот бит извлекает невалидную длительность передачи данных из отчета об ошибке подтверждения.	R	0
[14]	ack_with_err_14	Этот бит извлекает зарезервированные данные (которые зависят от прибора) из отчета об ошибке подтверждения.	R	0
[15]	ack_with_err_15	Этот бит извлекает нарушение протокола DSI из отчета об ошибке подтверждения.	R	0
[16]	dphy_errors_0	Этот бит указывает на ошибку ErrEsc от тракта 0.	R	0
[17]	dphy_errors_1	Этот бит указывает на ошибку синхронизации передачи данных в режиме низкого питания ErrSyncEsc от тракта 0.	R	0
[18]	dphy_errors_2	Этот бит указывает на ошибку управления ErrControl от тракта 0.	R	0
[19]	dphy_errors_3	Этот бит указывает на конфликт LP0 ErrContentionLP0 от тракта 0.	R	0
[20]	dphy_errors_4	Этот бит указывает на конфликт LP1 ErrContentionLP1 от тракта 0.	R	0
[31:21]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.49 Регистр INT_ST1

Этот регистр содержит статусы источников прерываний, связанных с таймутами, ECC, CRC, размерами пакетов, EoTp, стандартным и DBI интерфейсами.

Поля регистра INT_ST1 описаны в Таблица 17.72.

Таблица 17.72. Регистр INT_ST1

Диапазон	Название	Описание	Тип доступа	Исходное значение
[0]	to_hs_tx	Этот бит указывает, что счетчик таймаута высокоскоростной передачи данных достиг окончания и обнаружен конфликт.	R	0
[1]	to_lp_rx	Этот бит указывает, что счетчик таймаута передачи данных в режиме низкого питания достиг окончания и обнаружен конфликт.	R	0
[2]	ecc_single_err	Этот бит указывает, что в полученном пакете обнаружена и исправлена единичная ошибка ECC.	R	0
[3]	ecc_milti_err	Этот бит указывает, что в полученном пакете обнаружена множественная ошибка ECC.	R	0
[4]	crc_err	Этот бит указывает, что в полезной нагрузке полученного пакета обнаружена ошибка CRC.	R	0
[5]	pkt_size_err	Этот бит указывает, что при получении пакета обнаружена ошибка размера пакета.	R	0
[6]	eopt_err	Этот бит указывает, что в конце входящей передачи от внешних устройств не обнаружен пакет EoTp (пакет окончания передачи).	R	0
[7]	dpi_pld_wr_err	Этот бит указывает, что в процессе сохранения пиксельной линии DPI буфер FIFO полезной нагрузки переполнился и сохраненные данные повредились. Условия существования: ((DSI_HOST_DATAINTERFACE == 2) ||	R	0

Диапазон	Название	Описание	Тип доступа	Исходное значение
		(DSI_HOST_DATAINTERFACE == 3) || (DSI_HOST_DATAINTERFACE == 4))==1.		
[8]	gen_cmd_wr_err	Этот бит указывает, что система пыталась произвести запись команды через стандартный интерфейс, и буфер FIFO оказался полон. Таким образом, команда не была записана.	R	0
[9]	gen_pld_wr_err	Этот бит указывает, что система пыталась записать данные полезной нагрузки через стандартный интерфейс и буфер FIFO оказался полон. Таким образом, данные полезной нагрузки не были записаны.	R	0
[10]	gen_pld_send_err	Этот бит указывает, что в процессе сборки пакета от стандартного интерфейса, буфер FIFO полезной нагрузки опустел и произошла отправка поврежденных данных.	R	0
[11]	gen_pld_rd_err	Этот бит указывает, что в процессе чтения данных через DCS, буфер FIFO полезной нагрузки опустел и данные, отправленные к интерфейсу, повреждены.	R	0
[12]	gen_pld_recev_err	Этот бит указывает, что в процессе обратного чтения пакета от стандартного интерфейса, буфер FIFO полезной нагрузки переполнился и полученные данные повреждены.	R	0
[13]	dbi_cmd_wr_err	Этот бит указывает, что система пыталась записать команду через интерфейс DBI, но командный буфер FIFO оказался полон. Таким образом, команда не была записана. Условия существования: ((DSI_HOST_DATAINTERFACE == 1) || (DSI_HOST_DATAINTERFACE == 3))==1.	R	0
[14]	dbi_pld_wr_err	Этот бит указывает, что система пыталась записать данные полезной нагрузки через интерфейс DBI и буфер FIFO оказался полон. Таким образом, команда не была записана. Условия существования: ((DSI_HOST_DATAINTERFACE == 1) || (DSI_HOST_DATAINTERFACE == 3))==1.	R	0
[15]	dbi_pld_rd_err	Этот бит указывает, что в процессе чтения данных от DCS, буфер FIFO полезной нагрузки опустел и данные, отправленные к интерфейсу, повреждены. Условия существования: ((DSI_HOST_DATAINTERFACE == 1) || (DSI_HOST_DATAINTERFACE == 3))==1.	R	0
[16]	dbi_pld_recv_err	Этот бит указывает, что в процессе обратного чтения пакета интерфейса DBI, буфер FIFO полезной нагрузки переполнился и полученные данные повреждены. Условия существования: ((DSI_HOST_DATAINTERFACE == 1) || (DSI_HOST_DATAINTERFACE == 3))==1.	R	0
[17]	dbi_ilegal_comm_err	Этот бит указывает, что совершена попытка произвести запись запрещенной команды в интерфейс DBI и система заблокирована передачей. Условия существования: ((DSI_HOST_DATAINTERFACE == 1) || (DSI_HOST_DATAINTERFACE == 3))==1.	R	0

Диапазон	Название	Описание	Тип доступа	Исходное значение
[31:18]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.50 Регистр INT_MSK0

Этот регистр конфигурирует маски для источников прерываний, которые затрагивают регистр INT_ST0. Для демаскировки каждого отчета об ошибке следует записать "1" в соответствующее поле.

Поля регистра INT_MSK0 описаны в Таблица 17.73.

Таблица 17.73. Регистр INT_MSK0

Диапазон	Название	Описание	Тип доступа	Исходное значение
[0]	mask_ack_with_err_0	Маска для ack_with_err_0.	R/W	0
[1]	mask_ack_with_err_1	Маска для ack_with_err_1.	R/W	0
[2]	mask_ack_with_err_2	Маска для ack_with_err_2.	R/W	0
[3]	mask_ack_with_err_3	Маска для ack_with_err_3.	R/W	0
[4]	mask_ack_with_err_4	Маска для ack_with_err_4.	R/W	0
[5]	mask_ack_with_err_5	Маска для ack_with_err_5.	R/W	0
[6]	mask_ack_with_err_6	Маска для ack_with_err_6.	R/W	0
[7]	mask_ack_with_err_7	Маска для ack_with_err_7.	R/W	0
[8]	mask_ack_with_err_8	Маска для ack_with_err_8.	R/W	0
[9]	mask_ack_with_err_9	Маска для ack_with_err_9.	R/W	0
[10]	mask_ack_with_err_10	Маска для ack_with_err_10.	R/W	0
[11]	mask_ack_with_err_11	Маска для ack_with_err_11.	R/W	0
[12]	mask_ack_with_err_12	Маска для ack_with_err_12.	R/W	0
[13]	mask_ack_with_err_13	Маска для ack_with_err_13.	R/W	0
[14]	mask_ack_with_err_14	Маска для ack_with_err_14.	R/W	0
[15]	mask_ack_with_err_15	Маска для ack_with_err_15.	R/W	0
[16]	mask_dphy_errors_0	Маска для dphy_errors_0.	R/W	0
[17]	mask_dphy_errors_1	Маска для dphy_errors_1.	R/W	0

Диапазон	Название	Описание	Тип доступа	Исходное значение
[18]	mask_dphy_errors_2	Маска для dphy_errors_2.	R/W	0
[19]	mask_dphy_errors_3	Маска для dphy_errors_3.	R/W	0
[20]	mask_dphy_errors_4	Маска для dphy_errors_4.	R/W	0
[31:21]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.51 Регистр INT_MSK1

Этот регистр конфигурирует маски для источников прерываний, которые затрагивают регистр INT_ST1.

Поля регистра INT_MSK1 описаны в Таблица 17.74.

Таблица 17.74. Регистр INT_MSK1

Диапазон	Название	Описание	Тип доступа	Исходное значение
[0]	mask_to_hs_tx	Маска для to_hs_tx.	R/W	0
[1]	mask_to_lp_rx	Маска для to_lp_rx.	R/W	0
[2]	mask_ecc_single_err	Маска для ecc_single_err.	R/W	0
[3]	mask_ecc_milti_err	Маска для ecc_milti_err.	R/W	0
[4]	mask_crc_err	Маска для crc_err.	R/W	0
[5]	mask_pkt_size_err	Маска для pkt_size_err.	R/W	0
[6]	mask_eopt_err	Маска для eopt_err.	R/W	0
[7]	mask_dpi_pld_wr_err	Маска для dpi_pld_wr_err. Условия существования: ((DSI_HOST_DATAINTERFACE == 2) || (DSI_HOST_DATAINTERFACE == 3) || (DSI_HOST_DATAINTERFACE == 4))==1.	R/W	0
[8]	mask_gen_cmd_wr_err	Маска для gen_cmd_wr_err.	R/W	0
[9]	mask_gen_pld_wr_err	Маска для gen_pld_wr_err.	R/W	0
[10]	mask_gen_pld_send_err	Маска для gen_pld_send_err.	R/W	0
[11]	mask_gen_pld_rd_err	Маска для gen_pld_rd_err.	R/W	0
[12]	mask_gen_pld_recev_err	Маска для gen_pld_recev_err.	R/W	0
[13]	mask_dbi_cmd_wr_err	Маска для dbi_cmd_wr_err. Условия существования: ((DSI_HOST_DATAINTERFACE == 1) || (DSI_HOST_DATAINTERFACE == 3))==1.	R/W	0

Диапазон	Название	Описание	Тип доступа	Исходное значение
[14]	mask_dbi_pld_wr_err	Маска для dbi_pld_wr_err. Условия существования: ((DSI_HOST_DATAINTERFACE == 1) || (DSI_HOST_DATAINTERFACE == 3))==1.	R/W	0
[15]	mask_dbi_pld_rd_err	Маска для dbi_pld_rd_err. Условия существования: ((DSI_HOST_DATAINTERFACE == 1) || (DSI_HOST_DATAINTERFACE == 3))==1.	R/W	0
[16]	mask_dbi_pld_recv_err	Маска для dbi_pld_recv_err. Условия существования: ((DSI_HOST_DATAINTERFACE == 1) || (DSI_HOST_DATAINTERFACE == 3))==1.	R/W	0
[17]	mask_dbi_ilegal_comm_err	Маска для dbi_ilegal_comm_err. Условия существования: ((DSI_HOST_DATAINTERFACE == 1) || (DSI_HOST_DATAINTERFACE == 3))==1.	R/W	0
[31:18]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.52 Регистр PHY_CAL

Этот регистр управляет калибровкой фазового сдвига D-PHY.

Поля регистра PHY_CAL описаны в Таблица 17.75.

Таблица 17.75. Регистр PHY_CAL

Диапазон	Название	Описание	Тип доступа	Исходное значение
[0]	txskewcalhs	Когда бит txskewcalhs установлен в активное состояние, запускается высокоскоростная калибровка фазового сдвига (предполагается, что PHY находится в состоянии остановки).	R/W	0
[31:1]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.53 Регистр INT_FORCE0

Этот регистр назначает прерывания, которые затрагивают регистр INT_ST0.

Поля регистра INT_FORCE0 описаны в Таблица 17.76.

Таблица 17.76. Регистр INT_FORCE0

Диапазон	Название	Описание	Тип доступа	Исходное значение
[0]	force_ack_with_err_0	Назначает ack_with_err_0.	W	0
[1]	force_ack_with_err_1	Назначает ack_with_err_1.	W	0
[2]	force_ack_with_err_2	Назначает ack_with_err_2.	W	0
[3]	force_ack_with_err_3	Назначает ack_with_err_3.	W	0
[4]	force_ack_with_err_4	Назначает ack_with_err_4.	W	0
[5]	force_ack_with_err_5	Назначает ack_with_err_5.	W	0

Диапазон	Название	Описание	Тип доступа	Исходное значение
[6]	force_ack_with_err_6	Назначает ack_with_err_6.	W	0
[7]	force_ack_with_err_7	Назначает ack_with_err_7.	W	0
[8]	force_ack_with_err_8	Назначает ack_with_err_8.	W	0
[9]	force_ack_with_err_9	Назначает ack_with_err_9.	W	0
[10]	force_ack_with_err_10	Назначает ack_with_err_10.	W	0
[11]	force_ack_with_err_11	Назначает ack_with_err_11.	W	0
[12]	force_ack_with_err_12	Назначает ack_with_err_12.	W	0
[13]	force_ack_with_err_13	Назначает ack_with_err_13.	W	0
[14]	force_ack_with_err_14	Назначает ack_with_err_14.	W	0
[15]	force_ack_with_err_15	Назначает ack_with_err_15.	W	0
[16]	force_dphy_errors_0	Назначает dphy_errors_0.	W	0
[17]	force_dphy_errors_1	Назначает dphy_errors_1.	W	0
[18]	force_dphy_errors_2	Назначает dphy_errors_2.	W	0
[19]	force_dphy_errors_3	Назначает dphy_errors_3.	W	0
[20]	force_dphy_errors_4	Назначает dphy_errors_4.	W	0
[31:21]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.54 Регистр INT_FORCE1

Этот регистр назначает прерывания, которые затрагивают регистр INT_ST1.

Поля регистра INT_FORCE1 описаны в Таблица 17.77.

Таблица 17.77. Регистр INT_FORCE1

Диапазон	Название	Описание	Тип доступа	Исходное значение
[0]	force_to_hs_tx	Назначает to_hs_tx.	W	0
[1]	force_to_lp_rx	Назначает to_lp_rx.	W	0
[2]	force_ecc_single_err	Назначает ecc_single_err.	W	0
[3]	force_ecc_milti_err	Назначает ecc_milti_err.	W	0

Диапазон	Название	Описание	Тип доступа	Исходное значение
[4]	force_crc_err	Назначает crc_err.	W	0
[5]	force_pkt_size_err	Назначает pkt_size_err.	W	0
[6]	force_eopt_err	Назначает eopt_err.	W	0
[7]	force_dpi_pld_wr_err	Назначает dpi_pld_wr_err. Условия существования: ((DSI_HOST_DATAINTERFACE == 2) ||(DSI_HOST_DATAINTERFACE == 3) ||(DSI_HOST_DATAINTERFACE == 4))==1.	W	0
[8]	force_gen_cmd_wr_err	Назначает gen_cmd_wr_err.	W	0
[9]	force_gen_pld_wr_err	Назначает gen_pld_wr_err.	W	0
[10]	force_gen_pld_send_err	Назначает gen_pld_send_err.	W	0
[11]	force_gen_pld_rd_err	Назначает gen_pld_rd_err.	W	0
[12]	force_gen_pld_recev_err	Назначает gen_pld_recev_err.	W	0
[13]	force_dbi_cmd_wr_err	Назначает dbi_cmd_wr_err. Условия существования: ((DSI_HOST_DATAINTERFACE == 1) ||(DSI_HOST_DATAINTERFACE == 3))==1.	W	0
[14]	force_dbi_pld_wr_err	Назначает dbi_pld_wr_err. Условия существования: ((DSI_HOST_DATAINTERFACE == 1) || (DSI_HOST_DATAINTERFACE == 3))==1.	W	0
[15]	force_dbi_pld_rd_err	Назначает dbi_pld_rd_err. Условие существования: ((DSI_HOST_DATAINTERFACE == 1) || (DSI_HOST_DATAINTERFACE == 3))==1.	W	0
[16]	force_dbi_pld_recv_err	Назначает bi_pld_recv_err. Условия существования: ((DSI_HOST_DATAINTERFACE == 1) || (DSI_HOST_DATAINTERFACE == 3))==1.	W	0
[17]	force_dbi_ilegal_comm_err	Назначает dbi_ilegal_comm_err. Условия существования: ((DSI_HOST_DATAINTERFACE == 1) || (DSI_HOST_DATAINTERFACE == 3))==1.	W	0
[31:18]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.55 Регистр DSC_PARAMETER

Этот регистр конфигурирует сжатие потока диспеля (DSC).

Поля регистра DSC_PARAMETER описаны в Таблица 17.78.

Таблица 17.78. Регистр DSC_PARAMETER

Диапазон	Название	Описание	Тип доступа	Исходное значение
[0]	compression_mode	Когда установлено значение "1", этот бит подключает режим сжатия.	R/W	0
[7:1]	-	Зарезервировано для дальнейшего использования.		0
[9:8]	compress_algo	Это поле задает идентификатор алгоритма: 00 - VESA DSC Standard 1.1; 11 - алгоритм определяется поставщиком; 01, 10 - зарезервированы, не используются.	R/W	0
[15:10]	-	Зарезервировано для дальнейшего использования.		0
[17:16]	pps_sel	Это поле задает селектор PPS (набора параметров изображения): 00 = PPS таблица 1; 01 = PPS таблица 2; 10 = PPS таблица 3; 11 = PPS таблица 4.	R/W	0
[31:18]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.56 Регистр PHY_TMR_RD_CFG

Этот регистр конфигурирует временные интервалы, относящиеся к PHY, для выполнения некоторых операций в тактах байтового синхросигнала тракта.

Условия существования: ((DSI_HOST_DATAINTERFACE == 2) || (DSI_HOST_DATAINTERFACE == 3) || (DSI_HOST_DATAINTERFACE == 4))==1.

Поля регистра PHY_TMR_RD_CFG описаны в Таблица 17.79.

Таблица 17.79. Регистр PHY_TMR_RD_CFG

Диапазон	Название	Описание	Тип доступа	Исходное значение
[14:0]	max_rd_time	Это поле конфигурирует максимальное время, которое требуется для выполнения команды чтения в тактах байтового синхросигнала тракта. Этот регистр может быть модифицирован только в такой момент, когда ни одна команда чтения не выполняется.	R/W	0
[31:15]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.57 Регистр VID_SHADOW_CTRL

Этот регистр управляет функциями дублирования интерфейса DPI.

Условия существования: ((DSI_HOST_DATAINTERFACE == 2) || (DSI_HOST_DATAINTERFACE == 3) || (DSI_HOST_DATAINTERFACE == 4))==1

Поля регистра VID_SHADOW_CTRL описаны в Таблица 17.80.

Таблица 17.80. Регистр VID_SHADOW_CTRL

Диапазон	Название	Описание	Тип доступа	Исходное значение
[0]	vid_shadow_en	Когда установлено значение "1", интерфейс DPI получает активную конфигурацию от вспомогательных регистров. Когда эта функция установлена одновременно с vid_shadow_req, вспомогательные регистры обновляются автоматически.	R/W	0
[7:1]	-	Зарезервировано для дальнейшего использования.		0
[8]	vid_shadow_req	Когда установлено значение "1", этот бит выдает запрос на то, чтобы регистры dpi были скопированы из банка регистров во вспомогательные регистры. Когда запрос удовлетворен, этот бит автоматически обнуляется.	R/W	0
[15:9]	-	Зарезервировано для дальнейшего использования.		0
[16]	vid_shadow_pin_req	Когда установлено значение "1", внешний контакт выдает запрос на видео. В этом режиме vid_shadow_req игнорируется.	R/W	0
[31:17]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.58 Регистр DPI_VCID_ACT

Этот регистр хранит значение, которое контроллер использует для DPI_VCID.

Условия существования: ((DSI_HOST_DATAINTERFACE == 2) || (DSI_HOST_DATAINTERFACE == 3) || (DSI_HOST_DATAINTERFACE == 4))==1.

Поля регистра DPI_VCID_ACT описаны в Таблица 17.81.

Таблица 17.81. Регистр DPI_VCID_ACT

Диапазон	Название	Описание	Тип доступа	Исходное значение
[1:0]	dpi_vcid	Это поле задает идентификатор виртуального канала DPI, которым индексируются пакеты видеорежима.	R	0
[31:2]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.59 Регистр DPI_COLOR_CODING_ACT

Этот регистр хранит значение, которое контроллер использует для DPI_COLOR_CODING.

Условия существования: ((DSI_HOST_DATAINTERFACE == 2) || (DSI_HOST_DATAINTERFACE == 3) || (DSI_HOST_DATAINTERFACE == 4))==1.

Поля регистра DPI_COLOR_CODING_ACT описаны в Таблица 17.82.

Таблица 17.82. Регистр DPI_COLOR_CODING_ACT

Диапазон	Название	Описание	Тип доступа	Исходное значение
[3:0]	dpi_color_coding	Это поле задает цветовое кодирование DPI следующим образом: 0000: 16-битное, конфигурация 1; 0001: 16-битное, конфигурация 2; 0010: 16-битное, конфигурация 3; 0011: 18-битное, конфигурация 1; 0100: 18-битное, конфигурация 2; 0101: 24-битное; 0110: 20-битное, кодирование YCbCr 4:2:2, свободно упакованное (loosely packed); 0111: 24-битное YCbCr 4:2:2; 1000: 16-битное YCbCr 4:2:2; 1001: 30-битное; 1010: 36-битное; 1011: 12-битное YCbCr 4:2:0; 11xx: сжатые данные DSC24. Если выбран интерфейс eDPI и работа ведется в командном режиме (cmd_video_mode = 1), то: 0000: 16-битное, конфигурация 1; 0001: 16-битное, конфигурация 2; 0010: 16-битное, конфигурация 3; 0011: 18-битное, конфигурация 1; 0100: 18-битное, конфигурация 2; 0101-0111: 24-битное; 1Xxx: сжатые данные DSC24.	R	0
[7:4]	-	Зарезервировано для дальнейшего использования.		0
[8]	loosely18_en	Когда установлено значение "1", этот бит активирует свободно упакованный вариант для 18-битных конфигураций.	R	0
[31:9]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.60 Регистр DPI_LP_CMD_TIM_ACT

Этот регистр хранит значение, которое контроллер использует для DPI_LP_CMD_TIM.

Условия существования: ((DSI_HOST_DATAINTERFACE == 2) || (DSI_HOST_DATAINTERFACE == 3) ||

(DSI_HOST_DATAINTERFACE == 4))==1

Поля регистра DPI_LP_CMD_TIM_ACT описаны в Таблица 17.83.

Таблица 17.83. Регистр DPI_LP_CMD_TIM_ACT

Диапазон	Название	Описание	Тип доступа	Исходное значение
[7:0]	invact_lpcmd_time	Это поле используется для передачи команд в режиме низкого питания. Оно определяет в байтах размер наибольшего пакета, который может быть помещен на линию во время VACT (Vertical Active)-зоны.	R	0
[15:8]	-	Зарезервировано для дальнейшего использования.		0

Диапазон	Название	Описание	Тип доступа	Исходное значение
[23:16]	outvact_lp cmd_time	Это поле используется для передачи команд в режиме низкого питания. Оно определяет в байтах размер наибольшего пакета, который может быть помещен на линию во время VSA, VBP, и VFP-зон.	R	0
[31:24]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.61 Регистр VID_MODE_CFG_ACT

Этот регистр хранит значение, которое контроллер использует для VID_MODE_CFG.

Условия существования: ((DSI_HOST_DATAINTERFACE == 2) || (DSI_HOST_DATAINTERFACE == 3) || (DSI_HOST_DATAINTERFACE == 4))==1.

Поля регистра VID_MODE_CFG_ACT описаны в Таблица 17.84.

Таблица 17.84. Регистр VID_MODE_CFG_ACT

Диапазон	Название	Описание	Тип доступа	Исходное значение
[1:0]	vid_mode_ type	Это поле задает тип передачи при видеорежиме следующим образом: 00: немонопольный с синхронизирующими импульсами; 01: немонопольный с синхронизирующими событиями; 10 и 11: пакетный (монопольный) режим.	R	0
[2]	lp_vsa_en	Когда установлено значение "1", этот бит разрешает возвращение к режиму низкого питания в рамках периода VSA (vertical sync active), при условии, что временные периоды это позволяют.	R	0
[3]	lp_vbp_en	Когда установлено значение "1", этот бит разрешает возвращение к режиму низкого питания в рамках периода VBP (vertical back porch), при условии, что временные периоды это позволяют.	R	0
[4]	lp_vfp_en	Когда установлено значение "1", этот бит разрешает возвращение к режиму низкого питания в рамках периода VFP (vertical front porch), при условии, что временные периоды это позволяют.	R	0
[5]	lp_vact_en	Когда установлено значение "1", этот бит разрешает возвращение к режиму низкого питания в рамках периода VACT (vertical active), при условии, что временные периоды это позволяют.	R	0
[6]	lp_hbp_en	Когда установлено значение "1", этот бит разрешает возвращение к режиму низкого питания в рамках периода HBP (horizontal back porch), при условии, что временные периоды это позволяют.	R	0

Диапазон	Название	Описание	Тип доступа	Исходное значение
[7]	lp_hfp_en	Когда установлено значение "1", этот бит разрешает возвращение к режиму низкого питания в рамках периода HFP (horizontal front porch), при условии, что временные периоды это позволяют.	R	0
[8]	frame_bta_ack_en	Когда установлено значение "1", этот бит разрешает запрос на подтверждающий ответ в конце кадра.	R	0
[9]	lp_cmd_en	Когда установлено значение "1", этот бит разрешает передачу команд только в режиме низкого питания.	R	0
[31:10]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.62 Регистр VID_PKT_SIZE_ACT

Этот регистр хранит значение, которое контроллер использует для VID_PKT_SIZE.

Условия существования: ((DSI_HOST_DATAINTERFACE == 2) || (DSI_HOST_DATAINTERFACE == 3) || (DSI_HOST_DATAINTERFACE == 4))==1.

Поля регистра VID_PKT_SIZE_ACT описаны в Таблица 17.85.

Таблица 17.85. Регистр VID_PKT_SIZE_ACT

Диапазон	Название	Описание	Тип доступа	Исходное значение
[13:0]	vid_pkt_size	Это поле определяет число пикселей в одном видеопакете. - для 18-битных, не свободно упакованных типов данных, это число должно быть кратно 4; - для типов данных режима YCbCr, это число должно быть кратно двум, в соответствии с рекомендациями в спецификации DSI.	R	0
[31:14]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.63 Регистр VID_NUM_CHUNKS_ACT

Этот регистр хранит значение, которое контроллер использует для VID_NUM_CHUNKS.

Условия существования: ((DSI_HOST_DATAINTERFACE == 2) || (DSI_HOST_DATAINTERFACE == 3) || (DSI_HOST_DATAINTERFACE == 4))==1.

Поля регистра VID_NUM_CHUNKS_ACT описаны в Таблица 17.86.

Таблица 17.86. Регистр VID_NUM_CHUNKS_ACT

Диапазон	Название	Описание	Тип доступа	Исходное значение
[12:0]	vid_num_chunks	Этот регистр определяет число блоков, которые необходимо передать в течение периода Line (блок - это пара из видеопакета и пустого пакета). Если установлены значения "0" или "1", видеолиния по-прежнему передается одним пакетом. Если установлено значение "1", такой пакет является частью блока. Это значит, что за ним следует пустой пакет (при условии, что vid_null_size>0). В остальных случаях для передачи каждой видеолинии используется набор блоков.	R	0
[31:13]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.64 Регистр VID_NULL_SIZE_ACT

Этот регистр хранит значение, которое контроллер использует для VID_NULL_SIZE.

Условия существования: ((DSI_HOST_DATAINTERFACE == 2) || (DSI_HOST_DATAINTERFACE == 3) || (DSI_HOST_DATAINTERFACE == 4))==1.

Поля регистра VID_NULL_SIZE_ACT описаны в Таблица 17.87.

Таблица 17.87. Регистр VID_NULL_SIZE_ACT

Диапазон	Название	Описание	Тип доступа	Исходное значение
[12:0]	vid_null_size	Этот регистр определяет число байтов внутри пустого пакета. Установка значения "0" отключает пустые пакеты.	R	0
[31:13]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.65 Регистр VID_HSA_TIME_ACT

Этот регистр хранит значение, которое контроллер использует для VID_HSA_TIME.

Условия существования: ((DSI_HOST_DATAINTERFACE == 2) || (DSI_HOST_DATAINTERFACE == 3) || (DSI_HOST_DATAINTERFACE == 4))==1.

Поля регистра VID_HSA_TIME_ACT описаны в Таблица 17.88.

Таблица 17.88. Регистр VID_HSA_TIME_ACT

Диапазон	Название	Описание	Тип доступа	Исходное значение
[11:0]	vid_hsa_time	Это поле определяет период активной горизонтальной (строчной) синхронизации (HSA) в тактах байтового синхросигнала тракта.	R	0
[31:12]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.66 Регистр VID_HBP_TIME_ACT

Этот регистр хранит значение, которое контроллер использует для VID_HBP_TIME.

Условия существования: ((DSI_HOST_DATAINTERFACE == 2) || (DSI_HOST_DATAINTERFACE == 3) || (DSI_HOST_DATAINTERFACE == 4))==1.

Поля регистра VID_HBP_TIME_ACT описаны в Таблица 17.89.

Таблица 17.89. Регистр VID_HBP_TIME_ACT

Диапазон	Название	Описание	Тип доступа	Исходное значение
[11:0]	vid_hbp_time	Это поле определяет период горизонтальной (строчной) задней площадки гасящего импульса (HBP) в тактах байтового синхросигнала тракта.	R	0
[31:12]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.67 Регистр VID_HLINE_TIME_ACT

Этот регистр хранит значение, которое контроллер использует для VID_HLINE_TIME.

Условия существования: ((DSI_HOST_DATAINTERFACE == 2) || (DSI_HOST_DATAINTERFACE == 3) || (DSI_HOST_DATAINTERFACE == 4))==1.

Поля регистра VID_HLINE_TIME_ACT описаны в Таблица 17.90.

Таблица 17.90. Регистр VID_HLINE_TIME_ACT

Диапазон	Название	Описание	Тип доступа	Исходное значение
[14:0]	vid_hline_time	Это поле определяет длительность общего времени видеолинии (HSA+HBP+HACT+HFP) в величинах байтового синхросигнала тракта.	R	0
[31:15]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.68 Регистр VID_VSA_LINES_ACT

Этот регистр хранит значение, которое контроллер использует для VID_VSA_LINES.

Условия существования: ((DSI_HOST_DATAINTERFACE == 2) || (DSI_HOST_DATAINTERFACE == 3) || (DSI_HOST_DATAINTERFACE == 4))==1.

Поля регистра VID_VSA_LINES_ACT описаны в Таблица 17.91.

Таблица 17.91. Регистр VID_VSA_LINES_ACT

Диапазон	Название	Описание	Тип доступа	Исходное значение
[9:0]	vsa_lines	Это поле определяет период активной вертикальной (кадровой) синхронизации (VSA), измеряемый числом горизонтальных линий.	R	0
[31:10]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.69 Регистр VID_VBP_LINES_ACT

Этот регистр хранит значение, которое контроллер использует для VID_VBP_LINES.

Условия существования: ((DSI_HOST_DATAINTERFACE == 2) || (DSI_HOST_DATAINTERFACE == 3) || (DSI_HOST_DATAINTERFACE == 4))==1.

Поля регистра VID_VBP_LINES_ACT описаны в Таблица 17.92.

Таблица 17.92. Регистр VID_VBP_LINES_ACT

Диапазон	Название	Описание	Тип доступа	Исходное значение
[9:0]	vbp_lines	Это поле определяет период вертикальной (кадровой) задней площадки гасящего импульса (VBP), измеряемый числом горизонтальных линий.	R	0
[31:10]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.70 Регистр VID_VFP_LINES_ACT

Этот регистр хранит значение, которое контроллер использует для VID_VFP_LINES.

Условия существования: ((DSI_HOST_DATAINTERFACE == 2) || (DSI_HOST_DATAINTERFACE == 3) || (DSI_HOST_DATAINTERFACE == 4))==1.

Поля регистра VID_VFP_LINES_ACT описаны в Таблица 17.93.

Таблица 17.93. Регистр VID_VFP_LINES_ACT

Диапазон	Название	Описание	Тип доступа	Исходное значение
[9:0]	vfp_lines	Это поле определяет период вертикальной (кадровой) передней площадки гасящего импульса (VFP), измеряемый числом горизонтальных линий.	R	0
[31:10]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.71 Регистр VID_VACTIVE_LINES_ACT

Этот регистр хранит значение, которое контроллер использует для VID_VACTIVE_LINES.

Условия существования: ((DSI_HOST_DATAINTERFACE == 2) || (DSI_HOST_DATAINTERFACE == 3) || (DSI_HOST_DATAINTERFACE == 4))==1.

Поля регистра VID_VACTIVE_LINES_ACT описаны в Таблица 17.94.

Таблица 17.94. Регистр VID_VACTIVE_LINES_ACT

Диапазон	Название	Описание	Тип доступа	Исходное значение
[13:0]	v_active_lines	Это поле определяет период вертикальной активности (Vertical Active), измеряемый числом горизонтальных линий.	R	0
[31:14]	-	Зарезервировано для дальнейшего использования.		0

17.2.4.72 Регистр SDF_3D_ACT

Этот регистр хранит значение, которое контроллер использует для SDF_3D.

Условия существования: ((DSI_HOST_DATAINTERFACE == 2) || (DSI_HOST_DATAINTERFACE == 3) || (DSI_HOST_DATAINTERFACE == 4))==1.

Поля регистра SDF_3D_ACT описаны в Таблица 17.95.

Таблица 17.95. Регистр SDF_3D_ACT

Диапазон	Название	Описание	Тип доступа	Исходное значение
[1:0]	mode_3d	Это поле определяет, включен или отключен режим 3D и задает ориентацию дисплея. 00: 3D режим отключен (2D режим включен); 01: 3D режим включен, портретная ориентация; 10: 3D режим включен, альбомная ориентация; 11: зарезервировано.	R	0
[3:2]	format_3d	Это поле задает формат 3D изображения: 00: линия (чередование линий левого и правого потока данных); 01: кадр (чередование кадров левого и правого потока данных); 10: пиксель (чередование пикселей левого и правого потока данных); 11: зарезервировано.	R	0
[4]	second_vsync	Это поле определяет присутствие второго сигнала VSYNC между левым и правым изображениями, когда выбран формат 3D-изображения на основе кадра. 0: между левым и правым потоком данных нет синхронизирующих импульсов; 1: между левым и правым потоком данных подключен синхросигнал (HSYNC, VSYNC, импульс гашения).	R	0
[5]	right_first	Этот бит задает порядок подачи левого/правого изображения: 0: первым подается изображение с левого глаза, затем - с правого; 1: первым подается изображение с правого глаза, затем - с левого.	R	0
[15:6]	-	Зарезервировано для дальнейшего использования.		0
[16]	send_3d_cfg	Когда установлен, заставляет последующий VSS-пакет включать полезную нагрузку 3D-настроек в каждый VSS-пакет.	R	0
[31:17]	-	Зарезервировано для дальнейшего использования.		0

18. КЛАСТЕР СЛУЖЕБНОГО ПРОЦЕССОРА (RISC)

18.1 Общая информация

В состав группы блоков информационной безопасности и управления микросхемы К1892ВМ21Я входят:

- доверенное отечественное процессорное ядро «RISCore32» с MIPS32 совместимой архитектурой с дополнительными блоками (кластер Служебного процессора);
- доверенные таймеры;
- доверенная память;
- доверенный контроллер прерываний;
- доверенные блоки ввода/вывода на базе отечественного ядра MFBSP;
- доверенные блоки управления питанием.

Блоки информационной безопасности обеспечивают для микросхемы К1892ВМ21Я возможность проверки целостности программного обеспечения и контроля начальной загрузки.

Служебный процессор RISC, являющийся основой группы блоков информационной безопасности и управления микросхемы К1892ВМ21Я, реализован на базе отечественного процессорного ядра «RISCore32» и представляет собой кластер, который, кроме самого процессорного ядра, объединяет ряд служебных блоков.

В состав кластера входят: процессорное ядро «RISCore32» из библиотеки платформы «Мультикор», четырехканальный блок DMA память-память QDMA, блок регистров контроля кластера CSR и блок преобразования адресов VMMU (см. Раздел 20.5).

Блок VMMU используется, в частности, для трансляции адресов из 32-битного адресного пространства, в котором работают RISC и QDMA, в 40-битный физический адрес на уровне СнК.

Схема кластера RISC представлена на Рисунок 18.1.

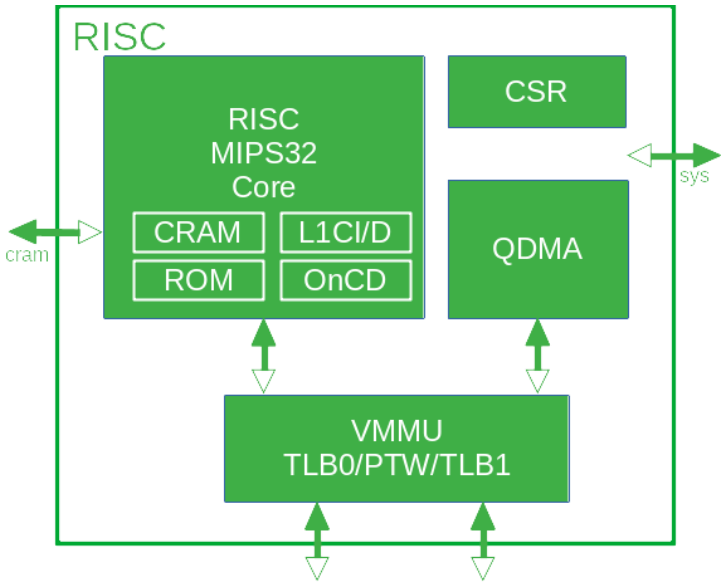


Рисунок 18.1. Схема кластера RISC

Непосредственно процессорное ядро «RISCore32» в составе кластера имеет следующие характеристики:

- Архитектура – MIPS32.
- Частота работы ядра - не менее 560 МГц при напряжении источника питания ядра (UCC2=0,9 В ±5%).
- 32-х битные пути передачи адреса и данных.
- Кэш команд объемом 32 Кбайта.
- Кэш данных объемом 32 Кбайта.
- Внутренняя память быстрого доступа CRAM объемом 32 КБ.
- Встроенная ROM память для начального загрузчика объемом 64 КБ для RISC0 и 32 Кбайта для RISC1.
- Регистры Count/Compare для прерываний реального времени.
- Отдельный вектор обработки исключений по прерываниям.
- Программируемое устройство управления памятью:
  - Два режима работы – с TLB и Fixed Mapped (FM);
  - 16 строк в режиме TLB;
  - В режиме FM адресные пространства отображаются с использованием битов регистров.
- Устройство умножения и деления.
- Поддержка отладки через JTAG (для RISC0).

- Блок OnCD с поддержкой отладки через регистры доступные в физической памяти.

18.1.1 Блок управления кластером RISC CSR

В состав кластера включены несколько служебных регистров для управления режимами работы процессорного ядра. Список этих регистров приведен в Таблица 18.1.

Таблица 18.1. Список регистров RISC CSR

Смещение	Обозначение	Назначение
Регистры управления и контроля VCPU		
0x00	CSR_VCPU	Регистр управления и состояния VCPU
0x04	EVENT_VCPU	Регистр управления сигналом события от VCPU

18.1.1.1 Регистр CSR_RISC

Таблица 18.2. Формат регистра CSR_RISC

Номер бита	Поле	Тип	Описание	Сброс
31:15	-	R/W	Резерв	0
14	INVALIDATE_D	R/W	При записи 1 в данный разряд кэш данных VCPU устанавливается в исходное состояние, то есть его содержимое девалидируется.	0
13	-	R/W	Резерв	0
12	INVALIDATE _I	R/W	При записи 1 в данный разряд кэш команд CPU устанавливается в исходное состояние, то есть его содержимое девалидируется.	0
11	TST_CACHE	R/W	Режим работы кэш программ и кэш данных: 0 – нормальный режим; 1 – режим тестирования. Используется только при технологическом тестировании кэш программ. Пользователям устанавливать этот режим запрещено.	0
10:2	-	R/W	Резерв	0
1	TR_CRAM	R/W	Режим размещения векторов прерываний при BEV = 0: 0 – вектора прерываний размещаются во внешней памяти (адреса типа 0x80000000); 1 – вектора прерываний размещаются во внутренней памяти CRAM (адреса типа 0xBFA0000 для Risc0 и 0xA3B0000 для Risc1);	0
0	FM	R/W	Режим преобразования виртуальных адресов VCPU в физические адреса: 0 – с использованием TLB; 1 – Fixed Mapped (FM).	1

18.1.1.2 Регистр EVENT_RISC

Таблица 18.3. Формат регистра EVENT_RISC

Номер бита	Поле	Тип	Описание	Сброс
31:15	-	R/W	Резерв	0

Номер бита	Поле	Тип	Описание	Сброс
0	EVENT	R/W	При записи 1 устанавливается сигнал события от VCPU, при записи 0 сигнал события снимается	0

18.2 ПРОЦЕССОРНОЕ ЯДРО КЛАСТЕРА RISC

18.2.1 Блок диаграмма

Блок схема процессорного ядра приведена на Рисунок 18.2.

Ядро содержит следующие узлы:

- устройство исполнения (execution core);
- устройство умножения и деления (mdu);
- системный управляющий сопроцессор (cp0);
- устройство управления памятью (mmu – memory management unit);
- контроллер кэш (cache controller);
- устройство шинного интерфейса (biu);
- кэш команд (i$);
- кэш данных (d$);
- средства отладки программ (oncd – on chip debugger) с jtag портом.

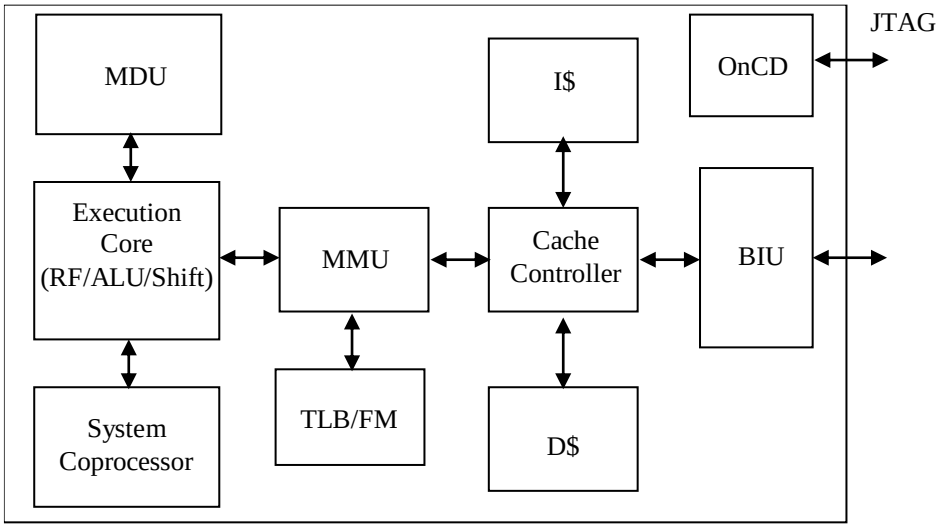


Рисунок 18.2. Блок схема процессорного ядра RISCorE32

18.2.2 Составляющие логические блоки

В следующих подразделах описываются устройства, входящие в состав процессорного ядра.

18.2.2.1 Устройство исполнения

Входящее в ядро устройство исполнения реализует архитектуру load-store (загрузка-сохранение) с однотактными операциями арифметического логического устройства (АЛУ) (логические операции, операции сдвига, сложение и вычитание). В ядре имеется тридцать два 32-х битных регистра общего назначения, используемых для скалярных целочисленных операций и вычисления адреса. В регистровом файле есть два порта чтения и один порт записи. Также используются обходные пути передачи данных для минимизации количества остановок конвейера.

В состав устройства исполнения входят:

- 32-х битный сумматор, используемый для вычисления адреса данных;
- адресное устройство для вычисления адреса следующей команды;
- логика определения перехода и вычисления адреса перехода;
- блок выравнивания при загрузке данных;
- мультиплексоры обходных путей передачи данных для избежания остановок конвейера в тех случаях, когда команды, производящие данные и команды, использующие эти данные расположены в программе достаточно близко;
- блок обнаружения нуля/единицы для реализации команд clz и clo;
- алу для выполнения побитных операций;
- сдвигающее устройство и устройство выравнивания при сохранении данных.

18.2.2.2 Устройство умножения/деления (MDU)

Устройство умножения/деления выполняет соответствующие операции. MDU выполняет операции умножения за 17 тактов, операции умножения с накоплением за 18 тактов, операции деления за 33 такта и операции деления с накоплением за 34 такта. Попытка активизировать следующую команду умножения/деления до завершения выполнения предыдущей, так же как и использование результата этой операции до того, как она закончена, вызывает остановку конвейера. В MDU имеется вывод, определяющий формат операции – знаковый или беззнаковый.

18.2.2.3 Системный управляющий сопроцессор

Сопроцессор отвечает за преобразование виртуального адреса в физический, протоколы кэш, систему управления исключениями, выбор режима функционирования (Kernel/User) и за разрешение/запрещение прерываний. Конфигурационная информация доступна посредством чтения регистров CP0 (см. Раздел 18.2.7 «Регистры CP0»).

18.2.2.4 Устройство управления памятью (MMU)

Процессорное ядро содержит устройство управления памятью (MMU), реализующее интерфейс между исполнительным блоком и контроллером кэш. Ядро может работать как в режиме TLB – с 16-строчной, полностью ассоциативной матрицей TLB, так и в режиме FM (Fixed Mapped), когда используются простые преобразования виртуального адреса в физический. Полностью устройство MMU описано в Разделе 18.2.2.4.

18.2.2.5 Контроллер кэш

CPU имеет кэш команд и кэш данных типа direct mapped объемом по 32 Кбайта. Кэш данных работает по протоколу write-through.

Кэш состоит из двух массивов – массива тэгов и массива данных. Кэш индексируется виртуально, поскольку для выбора соответствующей строки в обоих массивах используется виртуальный адрес. Это позволяет осуществлять доступ к кэш параллельно с преобразованием виртуального адреса в физический. Контроль осуществляется по физическому тэгу, так-так массив тэгов содержит физический, а не виртуальный адрес.

18.2.2.6 Устройство шинного интерфейса (BIU – Bus Interface Unit)

Устройство шинного интерфейса управляет внешними интерфейсными сигналами в соответствии со спецификацией шины AHB (Advanced High-performance Bus) архитектуры AMBA (Advanced Microcontroller Bus Architecture).

18.2.2.7 OnCD контроллер

В ядре имеется устройство для отладки программ OnCD с портом JTAG.

18.2.3 Конвейер

В RISC-ядре процессора реализован конвейер, состоящий из пяти стадий и аналогичный конвейеру ядра R3000. Конвейер дает возможность процессору работать на высокой частоте, при этом минимизируется сложность устройства, а также уменьшается стоимость и потребление энергии.

В этой главе содержатся следующие разделы:

- Раздел 18.2.3.1 «Стадии работы конвейера»;
- Раздел 18.2.3.7 «Операции умножения и деления»;
- Раздел 18.2.3.8 «Задержка выполнения команд перехода»;
- Раздел 18.2.3.9 «Обходные пути передачи данных (Data bypass)»;
- Раздел 18.2.3.10 «Задержка загрузки данных».

18.2.3.1 Стадии конвейера

Конвейер содержит пять стадий:

- выборка команды (стадия i- instruction);
- дешифрация команды (стадия d - data);
- исполнение команды (стадия e - execution);
- выборка из памяти (стадия m - memory);
- обратная запись (стадия w – write back).

На Рисунок 18.3 показаны операции, выполняемые RISC-ядром на каждом этапе конвейера.

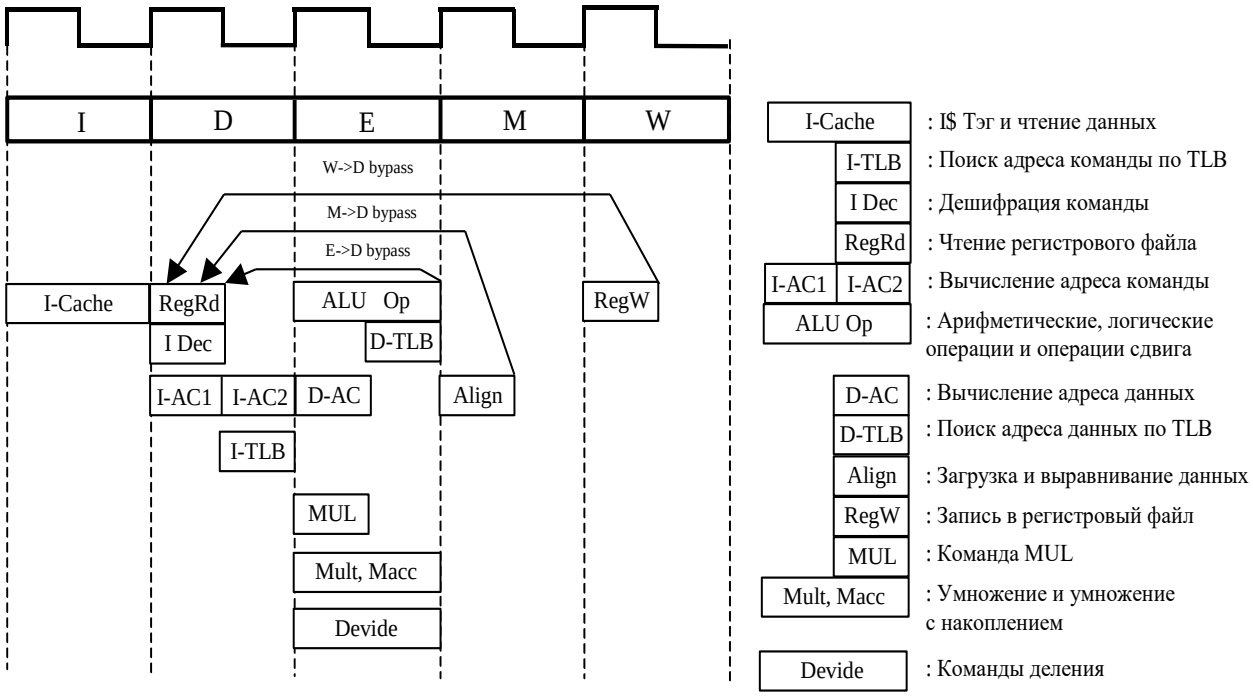


Рисунок 18.3.

18.2.3.2 Стадия I: выборка команды

На этой стадии команда выбирается из командного кэша.

18.2.3.3 Стадия D: дешифрация команды

На этой стадии:

- Операнды выбираются из регистрового файла.
- Операнды передаются на эту стадию со стадий E, M и W.
- ALU определяет, выполняется ли условие перехода и вычисляет виртуальный адрес перехода для команд перехода.
- Осуществляется преобразование виртуального адреса в физический.
- Производится поиск адреса команды по TLB и вырабатывается признак hit/miss.
- Командная логика выбирает адрес команды.

18.2.3.4 Стадия E: исполнение

На этой стадии:

- ALU выполняет арифметические или логические операции для команд типа регистр-регистр.
- Производится преобразование виртуального адреса в физический для данных, используемых командами загрузки и сохранения.
- Производится поиск данных по TLB и вырабатывается признак hit/miss.
- Все операции умножения и деления выполняются на этой стадии.

18.2.3.5 Стадия M: выборка из памяти

На этой стадии осуществляется загрузка и выравнивание загруженных данных в границах слова.

18.2.3.6 Стадия W: обратная запись

На этой стадии для команд типа регистр-регистр или для команд загрузки результат записывается обратно в регистровый файл.

18.2.3.7 Операции умножения и деления

Время выполнения этих операций соответствует 17 тактам для команд умножения и 18 тактам для команд умножения с накоплением, а также 33 тактам для команд деления и 34 тактам для команд деления с накоплением.

18.2.3.8 Задержка выполнения команд перехода (Jump, Branch)

Конвейер осуществляет выполнение команд перехода с задержкой в один такт. Однотактная задержка является результатом функционирования логики, ответственной за принятие решения о переходе на стадии D конвейера. Эта задержка позволяет использовать адрес перехода, вычисленный на предыдущей стадии, для доступа к команде на следующей D-стадии. Слот задержки перехода (branch delay slot) позволяет отказаться от остановок конвейера при переходе. Вычисление адреса и проверка условия перехода выполняются одновременно на стадии D. Итоговое значение PC (счетчика команд) используется для выборки очередной команды на стадии I, которая является второй командой после перехода. На Рисунок 18.4 показан слот задержки перехода.

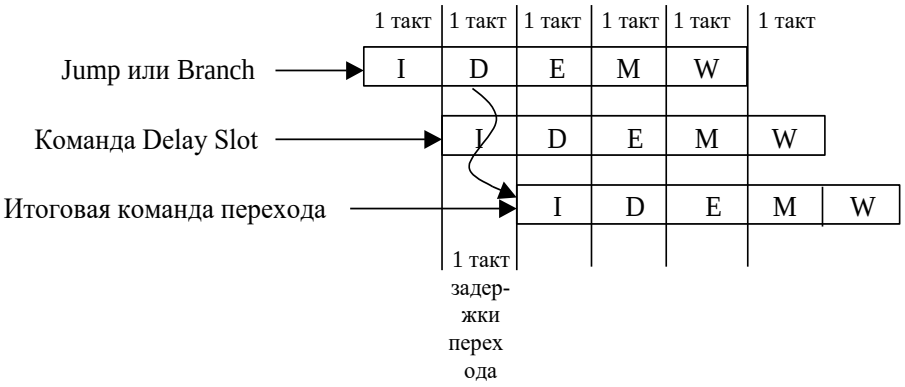


Рисунок 18.4. Слот задержки перехода

18.2.3.9 Обходные пути передачи данных (Data bypass)

Для большинства команд MIPS32 исходными операндами являются значения, хранящиеся в регистрах общего назначения. Эти операнды выбираются из регистрового файла в первой половине D-стадии. После исполнения на ALU результат, в принципе, готов для использования другими командами. Но запись результата в регистровый файл осуществляется только на стадии W. Это лишает следующую команду возможности использовать результат в течение 3-х циклов, если ее операндом является результат выполнения последней операции, сохраненный в регистровом файле. Для преодоления этой проблемы используются обходные пути передачи данных.

Мультиплексоры обходных путей передачи данных для обоих операндов располагаются между регистровым файлом и ALU (Рисунок 18.5). Они позволяют передавать данные с выхода стадий E, M и W конвейера прямо на стадию D, если один из регистров источника (source) декодируемой команды совпадает с регистром назначения (target) одной из предшествующих команд. Входы мультиплексоров подключены к обходным путям M→D и E→D, а также W→D.

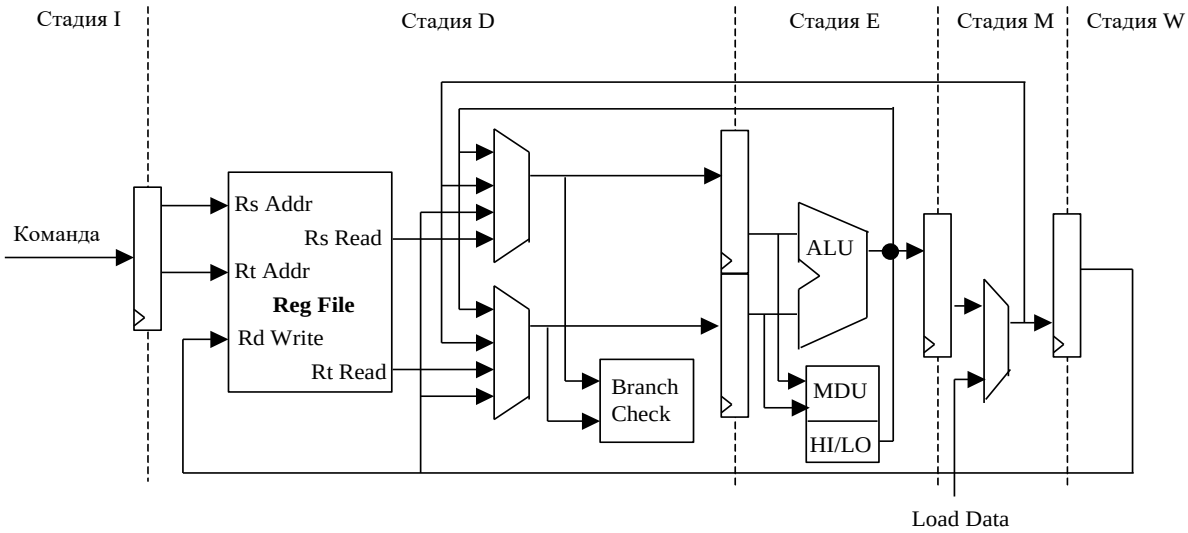


Рисунок 18.5.

На Рисунок 18.6 показаны обходные пути передачи данных для команды Add_1, за которой следует команда Sub_2 и затем снова Add_3. Поскольку команда Sub_2 в качестве одного из операндов использует результат операции Add_1, используется обходной путь E→D. Следующая команда Add_3 использует результаты обеих предшествующих операций: Add_1 и Sub_2. Так как данные команды Add_1 в это время находятся на стадии M, используется обходной путь M→D. Кроме того, вновь используется обходной путь E→D для передачи результата операции Sub_2 команде Add_3.

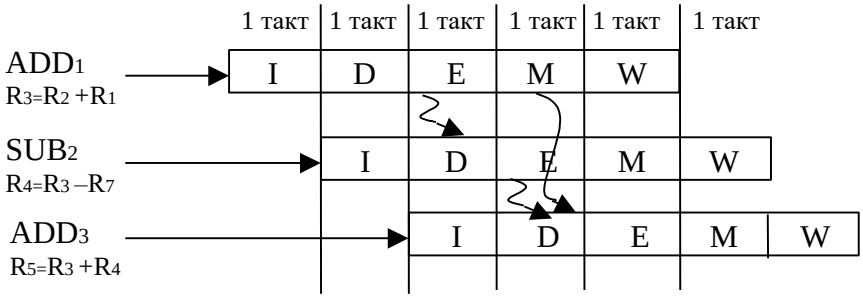


Рисунок 18.6.

18.2.3.10 Задержка загрузки данных

Данные, выбираемые командами загрузки (Load), становятся доступными на конвейере только после выравнивания на стадии M. При этом данные, являющиеся исходными операндами, должны предоставляться командам для обработки уже на стадии D. Поэтому, если сразу за командой загрузки следует команда, для которой один из регистров исходных операндов совпадает с регистром, в который производится загрузка данных, это вызывает приостановку в работе конвейера на стадии D. Эта приостановка осуществляется аппаратной вставкой команды NOP. Во время этой задержки часть конвейера, которая находится дальше стадии D, продолжает продвигаться. Если же команда, использующая загружаемые данные, следует за командой загрузки не сразу, а через одну или через две, то для обеспечения бесперебойной работы конвейера используется один из обходных путей передачи данных: M→D или W→D (Рисунок 18.7).

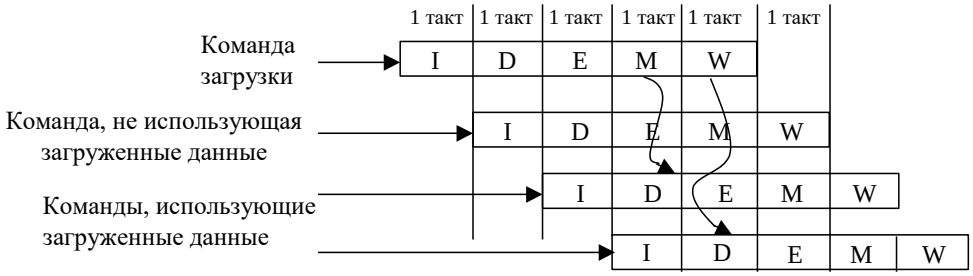


Рисунок 18.7.

18.2.4 Устройство управления памятью (MMU)

18.2.4.1 Введение

Процессорное ядро содержит устройство управления памятью (MMU), реализующее интерфейс между устройством исполнения и контроллером кэш. MMU преобразует виртуальный адрес в физический прежде, чем посылает запрос контроллеру кэш для сравнения тэга или блоку шинного интерфейса для доступа к внешнему запоминающему устройству. Это преобразование является очень полезным свойством функционирования операционных систем при управлении физической памятью таким образом, чтобы в ней размещались несколько процессов, активных в одной и той же области памяти, и может быть даже на одном виртуальном адресе, но обязательно в различных областях физической памяти. Другие свойства MMU – защита зон памяти и определение протокола кэш.

MMU может выполнять преобразование адресов в двух режимах: в режиме TLB и в режиме FM. Режим преобразования определяется внешним сигналом.

В режиме TLB используется полностью ассоциативная таблица преобразования адресов (TLB), имеющая 16 парных строк (entries). Во время преобразования осуществляется поиск соответствия по TLB. Если искомая строка отсутствует, генерируется прерывание.

В режиме FM (Fixed Mapped) работа MMU основана на простом алгоритме, обеспечивающем преобразование виртуальнго адреса в физический посредством механизма фиксированного отображения. Правила преобразования отличаются для различных областей виртуального адресного пространства (useg/kuseg, kseg0, kseg1, kseg2, kseg3).

На Рисунок 18.8 показано, взаимодействие MMU с процедурой доступа к кэш в режиме TLB, а на Рисунок 18.9 – в режиме FM.

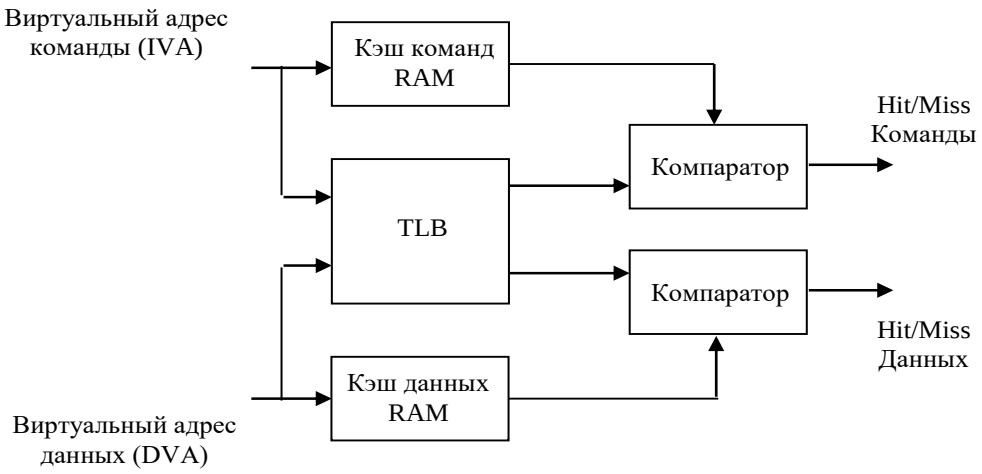


Рисунок 18.8.

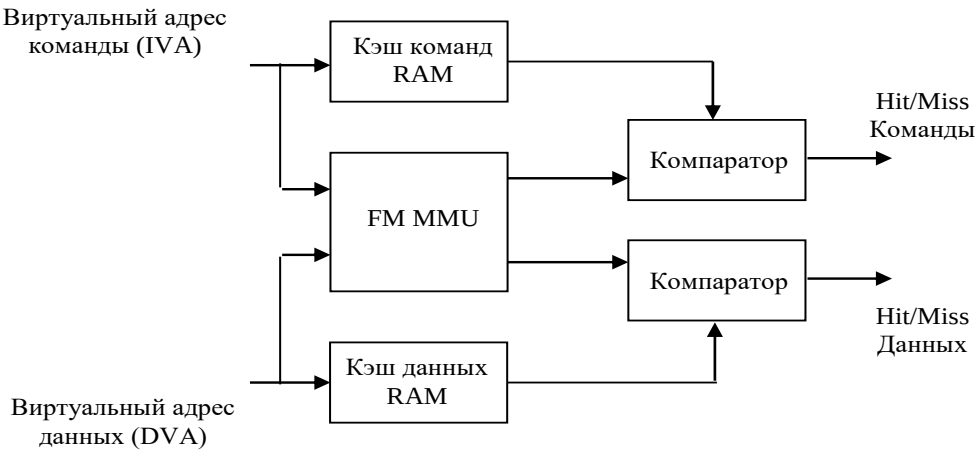


Рисунок 18.9.

18.2.4.2 Режимы работы.

Процессорное ядро поддерживает два режима работы:

- режим user (непривилегированный режим);
- режим kernel (привилегированный режим).

Режим User в основном используется для прикладных программ. Режим Kernel обычно используется для обработки исключительных ситуаций и привилегированных функций операционной системы, включая управление сопроцессором CP0 и доступ к устройствам ввода-вывода.

Преобразования, выполняемые MMU, зависят от режима работы процессора.

18.2.4.3 Виртуальные сегменты памяти

Виртуальные сегменты памяти, на которые делится адресное пространство, различаются в зависимости от режима работы процессора. На Рисунок 18.10 показана сегментация для 4 ГБайт (2^{32} байт) виртуального адресного пространства, адресуемого 32-разрядным виртуальным адресом для обоих режимов работы.

Ядро входит в режим Kernel после аппаратного сброса или когда происходит исключение. В режиме Kernel программное обеспечение имеет доступ к полному адресному пространству и ко всем регистрам CP0. В режиме User доступ ограничен подмножеством виртуального адресного пространства (0x_0000 - 0x7FFF_FFFF) и запрещен доступ к функциям CP0. В режиме User недоступны виртуальные адреса 0x8000_0000 - 0xFFFF_FFFF и обращение к ним вызывает исключение.

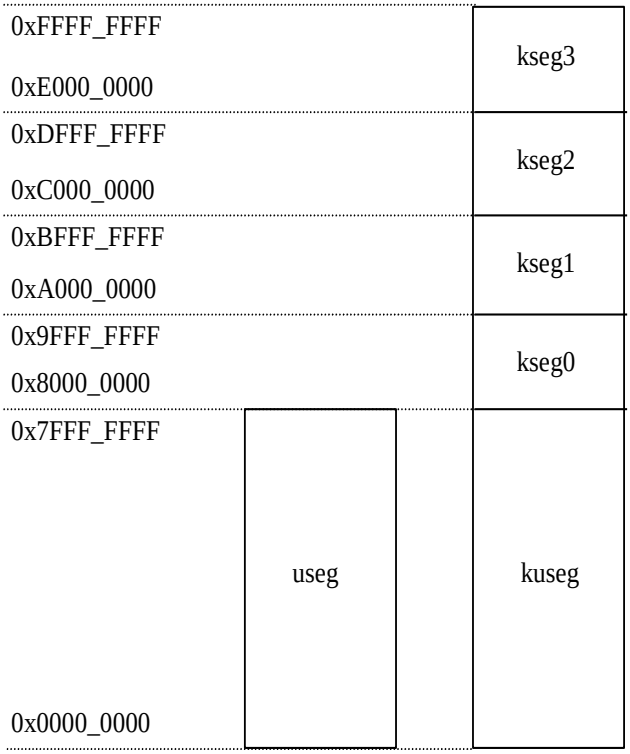


Рисунок 18.10. Карта виртуальной памяти для режимов User и Kernel

Каждый из сегментов, показанных на рис. 3.3, является либо отображаемым (mapped), либо неотображаемым (unmapped). Различие объясняется в следующих двух разделах. Далее, в Разделах 3.2.2 «Режим User» и 3.2.3 «Режим Kernel» говорится, какие именно сегменты являются отображаемыми, а какие – нет.

18.2.4.3.1 Неотображаемые сегменты

В неотображаемом сегменте механизмы TLB или FM для преобразования виртуального адреса в физический не используются. Особенно важно иметь неотображаемые сегменты памяти после аппаратного сброса, потому что TLB еще не запрограммировано и не может осуществлять преобразования.

Для неотображаемых сегментов преобразование виртуального адреса в физический является фиксированным.

Все неотображаемые сегменты, за исключением kseg0, никогда не кэшируемы. Кэшируемость kseg0 определяется полем K0 регистра Config CP0.

18.2.4.3.2 Отображаемые сегменты

В отображаемом сегменте для преобразования виртуального адреса в физический используются TLB или FM. Преобразование отображаемых сегментов имеет постраничную основу. При преобразовании выявляется информация о кэшируемости страницы, а также атрибуты защиты, относящиеся к странице.

Для режима FM отображаемые сегменты имеют закрепленное преобразование виртуального адреса в физический. Кэшируемость сегмента определяется значениями полей K23 и KU регистра Config CP0. При FM-преобразовании невозможна защита сегментов от записи.

18.2.4.4 Режим User

В режиме User доступно однородное виртуальное адресное пространство размером 2 ГБайта, называемое сегментом пользователя.

На Рисунок 18.11 показано размещение виртуального адресного пространства режима User.

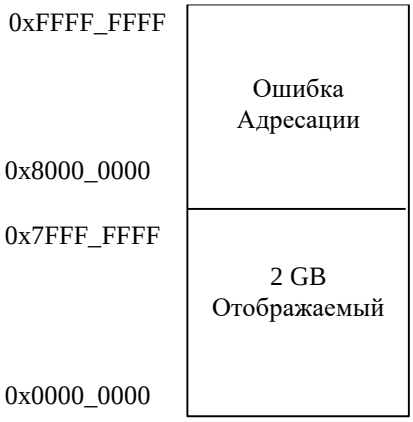


Рисунок 18.11.

Сегмент потребителя начинается с адреса 0x_0000 и заканчивается адресом 0x7FFF_FFFF. Обращения по всем остальным адресам вызывают прерывания по ошибке адресации.

Процессор находится в режиме User, если в регистре Status CP0 установлены следующие значения разрядов:

- UM = 1
- EXL = 0
- ERL = 0

В Таблица 18.4 приводятся характеристики сегмента useg режима User.

Таблица 18.4.

Адрес	Регистр Состояния			Имя сегмента	Диапазон адресов	Размер сегмента
	EXL	ERL	UM			
A(31)=0	0	0	1	useg	0x_0000 → 0x7FFF_FFFF	2GB (2^{31} байт)

Для всех допустимых виртуальных адресов режима User старший значащий бит адреса равен нулю, поскольку в режиме User допустимо обращение только к нижней половине

карты виртуальной памяти. Любая попытка обращения по адресу со старшим битом, равным 1, в режиме User вызывает прерывание по ошибке адресации.

Система отображает все ссылки к useg через TLB или FM. В режиме TLB виртуальный адрес перед преобразованием расширяется содержимым 8-разрядного поля ASID, образуя уникальный виртуальный адрес. Кэшируемость ссылки для страницы в этом режиме определяется установкой определенных бит строки TLB. В режиме FM кэшируемость задается полем KU Регистра Конфигурации CP0.

18.2.4.5 Режим Kernel

Процессор находится в режиме Kernel, когда регистр Status CP0 содержит хотя бы одно из следующих значений:

- UM = 0
- ERL = 1
- EXL = 1

Когда обнаруживается исключение, биты EXL или ERL устанавливаются, и процессор входит в режим Kernel. При завершении процедуры обработки исключения обычно выполняется команда возвращения из исключения (ERET). Команда ERET осуществляет переход по PC исключения, очищает ERL и EXL (если ERL=0). В результате возможен возврат процессора в режим User.

Виртуальное адресное пространство режима Kernel разделено на области в соответствии со значением старших битов виртуального адреса, как показано на Рисунок 18.12. Кроме того, в Таблица 18.5 содержатся характеристики сегментов режима Kernel.

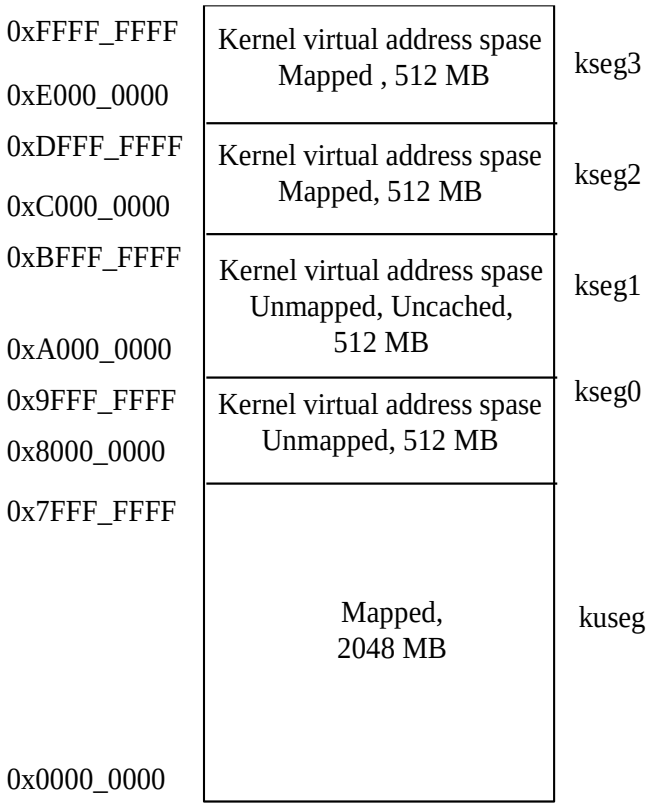


Рисунок 18.12.

Таблица 18.5.

Адрес	Регистр Состояния			Имя сегмента	Диапазон адресов	Размер сегмента
	EXL	ERL	UM			
A(31)=0	UM = 0			kuseg	0x_0000 → 0x7FFF_FFFF	2 GB (2^{31})
A(31:29)=100_2	или			kseg0	0x8000_0000 → 0x9FFF_FFFF	512 MB (2^{29})

A(31:29)=101_2	EXL=1 или ERL=1	kseg1	0xA000_0000 → 0xBFFF_FFFF	512 MB (2^{29})
A(31:29)=110_2		kseg2	0xC000_0000 → 0xDFFF_FFFF	512 MB (2^{29})
A(31:29)=111_2		kseg3	0xE000_0000 → 0xFFFF_FFFF	512 MB (2^{29})

18.2.4.5.1 Режим Kernel, Пространство пользователя (kuseg)

Если в режиме Kernel старший значащий бит виртуального адреса (A[31]) равен 0, выбирается 32-х разрядное виртуальное адресное пространство kuseg, полностью покрывающее 2^{31} байт (2 Гбайта) текущего адресного пространства пользователя, отображенного на адреса 0x_0000 - 0x7FFF_FFFF. В режиме TLB виртуальный адрес расширяется 8 битным значением поля ASID для образования уникального виртуального адреса. В режиме TLB, при ERL=0, кэшируемость определяется полем C строки TLB. В режиме FM кэшируемость определяется полем KU регистра Config CP0. Когда ERL = 1 в регистре Status, область адресов пользователя становится 2^{31}-байтовым неотображаемым и некэшируемым адресным пространством. При этом виртуальный адрес kuseg отображается на тот же физический адрес и не включает поля ASID.

18.2.4.5.2 Режим Kernel, пространство 0 режима Kernel (kseg0).

Если в режиме Kernel три старших бита виртуального адреса равны 100_2, выбирается 32-разрядное виртуальное адресное пространство kseg0. Это область размером 2^{29} байт (512 MB), которая расположена внутри границ, определяемых адресами 0x8000_0000 и 0x9FFF_FFFF. Ссылки к kseg0 не отображаются, а физический адрес получается вычитанием 0x8000_0000 из виртуального адреса. Кэшируемость сегмента kseg0 определяется значением поля K0 регистра Config CP0.

18.2.4.5.3 Режим Kernel, пространство 1 режима Kernel (kseg1)

Если в режиме Kernel три старших бита виртуального адреса равны 101_2, выбирается 32-разрядное виртуальное адресное пространство kseg1. Это область размером 2^{29} байт (512 MB), которая расположена внутри границ, определяемых адресами 0xA000_0000 и 0xBFFF_FFFF. Ссылки к kseg1 не отображаются, а физический адрес получается вычитанием 0xA000_0000 из виртуального адреса.

18.2.4.5.4 Режим Kernel, пространство 2 режима Kernel (kseg2)

Если в режиме Kernel UM = 0, ERL = 1, или EXL = 1 в регистре Status CP0, и три старших бита виртуального адреса равны 110_2, выбирается 32-разрядное виртуальное адресное пространство kseg2. В режиме FM это виртуальное пространство Kernel размером 2^{29} байт (512 Мбайт) зафиксировано в физических адресах 0xC000_0000 - 0xDFFF_FFFF и его кэшируемость определяется полем K23 Регистра Конфигурации. В режиме TLB это пространство отображается через TLB и его кэшируемость определяется полем C строки TLB.

18.2.4.5.5 Режим Kernel, пространство 3 режима Kernel (kseg3)

Если в режиме Kernel три старших бита виртуального адреса равны 111_2, выбирается 32-разрядное виртуальное адресное пространство kseg3. В режиме FM это виртуальное пространство Kernel размером 2^{29} байт (512 Мбайт) зафиксировано в физических адресах 0xE000_0000 - 0xFFFF_FFFF и его кэшируемость определяется полем K23 регистра Config. В режиме TLB это пространство отображается через TLB и его кэшируемость определяется полем C строки TLB.

18.2.4.6 Буфер быстрого преобразования адреса (TLB)

В этой главе описывается управление памятью с помощью буфера быстрого преобразования адреса (TLB), которое осуществляется в режиме TLB.

В режиме TLB реализуется полностью ассоциативный буфер быстрого преобразования адреса (TLB), содержащий 16 двойных строк, позволяющих отображать 32 виртуальных страницы в соответствующие физические адреса. TLB организовано в виде 16 парных строк – четных и нечетных, содержащих страницы размером от 4 Кбайт до 16 Мбайт, которые хранятся в 4-Гбайтном физическом адресном пространстве. Задача TLB состоит в преобразовании виртуальных адресов и их соответствующего идентификатора адресного пространства (ASID) в физический адрес памяти. Преобразование выполняется путем сравнения старших разрядов виртуального адреса (вместе с битами поля ASID) с каждой из строк тэговой порции TLB и иначе называется поиском соответствия по TLB (поиском соответствия тэга оной из строк виртуальному адресу на входе TLB).

Буфер TLB организован в виде страничных пар для минимизации общего количества хранящейся информации. Каждая строка тэговой порции соответствует двум физическим строкам данных – строке четных страниц и строке нечетных страниц. Самый старший разряд виртуального адреса, не участвующий в сравнении тэгов, определяет какая строка из двух строк данных используется. Поскольку размер страницы может варьироваться для каждой пары страниц, определение адресных разрядов, участвующих в сравнении и разряда, задающего четность страницы, должно осуществляться динамически при поиске по TLB.

На Рисунок 18.13 показано содержание одной из 16 двойных строк TLB.

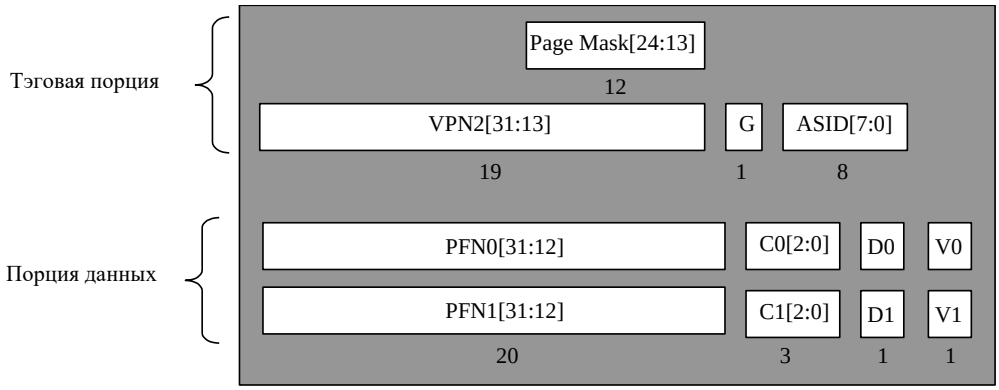


Рисунок 18.13.

Описание полей строки TLB приведены в Таблица 18.6.

Таблица 18.6.

<table>
<tr><th>Название поля</th><th>Описание</th></tr>
<tr><td>Page Mask[24:13]</td><td>Значение маски размера страницы. Определяет размер страницы маскировкой соответствующих разрядов VPN2, и тем самым исключением их из рассмотрения. Также используется для задания адресного разряда, определяющего четность страницы (PFN0-PFN1). См. следующую таблицу:
<table>
<tr><th>Page Mask[11:0]</th><th>Размер страницы</th><th>Бит определения четности</th></tr>
<tr><td>0000_0000_0000</td><td>4 КБ</td><td>VAddr[12]</td></tr>
<tr><td>0000_0000_0011</td><td>16 КБ</td><td>VAddr[14]</td></tr>
<tr><td>0000_0000_1111</td><td>64 КБ</td><td>VAddr[16]</td></tr>
<tr><td>0000_0011_1111</td><td>256 КБ</td><td>VAddr[18]</td></tr>
<tr><td>0000_1111_1111</td><td>1 МБ</td><td>VAddr[20]</td></tr>
<tr><td>0011_1111_1111</td><td>4 МБ</td><td>VAddr[22]</td></tr>
<tr><td>1111_1111_1111</td><td>16 МБ</td><td>VAddr[24]</td></tr>
</table>
В столбце Page Mask приведены все возможные значения Page Mask. Поскольку каждая пара битов этого поля всегда имеет одинаковое значение, физическая строка в TLB содержит сокращенную версию Page Mask, содержащую только 6 бит. Однако для программы это значение всегда преобразуется в 12-битное.</td></tr>
<tr><td>VPN2[31:13]</td><td>Виртуальный номер страницы, поделенный на 2. Данное поле содержит старшие разряды виртуального номера страницы. Виртуальный номер разделен на 2 потому, что он соответствует паре страниц TLB. Разряды 31:25 всегда участвуют в сравнении. Участие в сравнении разрядов 24:13 зависит от размера страницы, задаваемого полем Page Mask.</td></tr>
<tr><td>G</td><td>Бит глобальности. Если он установлен, данная строка является глобальной для всех процессов и подпроцессов, и таким образом, поле ASID исключается из рассмотрения.</td></tr>
<tr><td>ASID[7:0]</td><td>Идентификатор адресного пространства. Определяет процесс или подпроцесс, с которым ассоциируется данная строка TLB.</td></tr>
</table>

Продолжение Таблица 18.6.

<table>
<tr><th>Название поля</th><th>Описание</th></tr>
<tr><td>PFN0[31:12],
PFN0[31:12]</td><td>Физический номер кадра. Задает старшие разряды физического адреса. Для страниц размером более 4 Кбайта используется подмножество этого поля.</td></tr>
<tr><td>C0[2:0],
C1[2:0]</td><td>Кэшируемость. Содержит закодированное значение атрибута кэшируемости и определяет, должна ли страница помещаться в кэш или нет. Поле кодируется следующем образам:

C[2:0] Атрибуты когерентности
<table><tr><td>000</td><td>При записи преобразуется в код 011</td></tr><tr><td>001</td><td>При записи преобразуется в код 011</td></tr><tr><td>010</td><td>Некэшируемая страница</td></tr><tr><td>011</td><td>Кэшируемая страница</td></tr><tr><td>100</td><td>При записи преобразуется в код 011</td></tr><tr><td>101</td><td>При записи преобразуется в код 011</td></tr><tr><td>110</td><td>При записи преобразуется в код 011</td></tr><tr><td>111</td><td>При записи преобразуется в код 010</td></tr></table></td></tr>
<tr><td>D0,
D1</td><td>“Dirty” (Грязная страница) – бит разрешения записи. Показывает, что в страницу была сделана запись и/или разрешена запись в данную страницу. Если этот бит установлен, разрешены операции сохранения в данной странице. Если не установлен, сохранения в данной странице будут вызывать исключения модификации.</td></tr>
<tr><td>V0,
V1</td><td>Бит валидности. Показывает, что данная строка TLB и, соответственно, отображение виртуальной страницы, действительны. Если этот бит установлен, то обращения к данной странице разрешены. Если не установлен, то обращения к странице будут вызывать исключения инвалидности TLB (TLB invalid).</td></tr>
</table>

Для заполнения строки TLB используются команды TLBWI и TLBWR (См. Главу 7). Перед запуском этих команд нужно обновить некоторые регистры CP0, записав в них значения, которые будут затем помещены в строку TLB.

- Значение Page Mask задается в регистре Page Mask CP0.
- Значения VPN2 и ASID задаются в регистре EntryHi CP0.
- Значения PFN0, C0, D0, V0 и G задаются в регистре EntryLo0 CP0.
- Значения PFN1, C1, D1, V1 и G задаются в регистре EntryLo1 CP0.

Биты глобальности G входят в оба регистра EntryLo0 и EntryLo1. Бит G строки TLB является результатом логической операции И, проведенной над битами глобальности из EntryLo0 и EntryLo1. Более подробно эти регистры описаны в Главе 4.

Наличие идентификатора адресного пространства (ASID) дает возможность уменьшить частоту попаданий при поисках по TLB на контекстной основе. Это определяет возможность одновременного существования нескольких процессов как в TLB, так и в кэш команд. Значение ASID хранится в регистре EntryHi и сравнивается со значением ASID каждой строки.

18.2.4.7 Преобразование виртуального адреса в физический в режиме TLB.

Преобразование виртуального адреса в физический начинается со сравнения виртуального адреса на выходе процессора с виртуальными адресами, хранящимися в TLB. Соответствие имеет место, если виртуальный номер страницы (VPN) адреса совпадает с полем VPN строки TLB с учетом маски, хранящейся в этой строке, а также выполняется одно из двух условий:

- установлен бит глобальности (g) для четных и нечетных страниц в строке tlb;
- поле asid виртуального адреса совпадает с полем asid строки tlb.

Это соответствие называется попаданием TLB. Если не имеется ни одного соответствия, возникает исключение промаха TLB и программному обеспечению дается возможность пополнить TLB из расположенной в памяти таблицы страниц виртуальных /физических адресов. На Рисунок 18.14 показана логика преобразования виртуального адреса в физический.

На этом рисунке виртуальный адрес расширяется 8-разрядным идентификатором адресного пространства (ASID), который уменьшает частоту попаданий при просмотрах TLB на контекстной основе. Это 8-разрядное поле ASID содержит номер, присвоенный процессу, и хранится в регистре EntryHi CP0.

1. Виртуальный адрес (VA), представленный виртуальным номером страницы (VPN), сравнивается с тэгом из строки TLB (VPN2) с учетом маски (PageMask).

2. Если имеется соответствие, номер страничного кадра (PFN0 или PFN1, в зависимости от значения бита четности – самого старшего бита, не участвующего в сравнении) извлекается и помещается в старшие разряды физического адреса (PA)

3. В младшие разряды физического адреса помещается смещение (Offset), не участвующее в сравнении.

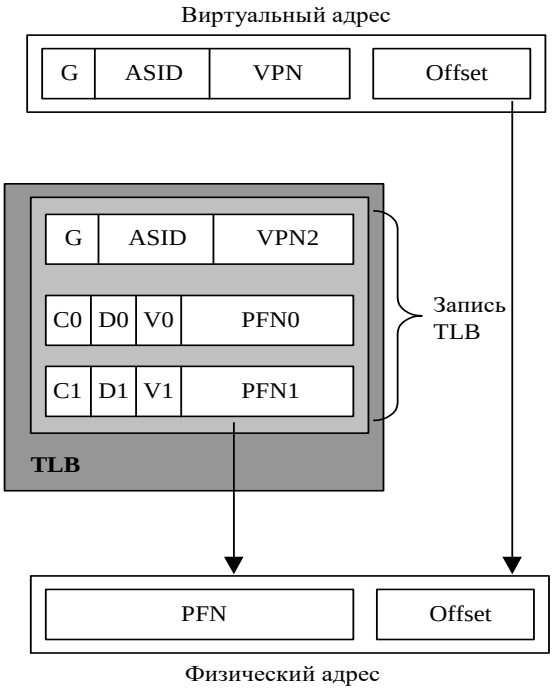


Рисунок 18.14.

Когда происходит совпадение виртуальных адресов при поиске по TLB, физический номер кадра (PFN) извлекается из соответствующей физической порции строки TLB и дополняется смещением, взятым из виртуального адреса, формируя таким образом физический адрес. Смещение представляет собой адрес в пределах пространства страничного кадра. Как показано на рисунке, смещение не пропускается через TLB.

На Рисунок 18.15 показана блок-схема процесса преобразования адреса. В верхней части рисунка показан виртуальный адрес для страницы размером 4 Кбайта. Ширина поля смещения определяется размером страницы.

В нижней части рисунка показан виртуальный адрес для страницы размером 16 Мбайт.

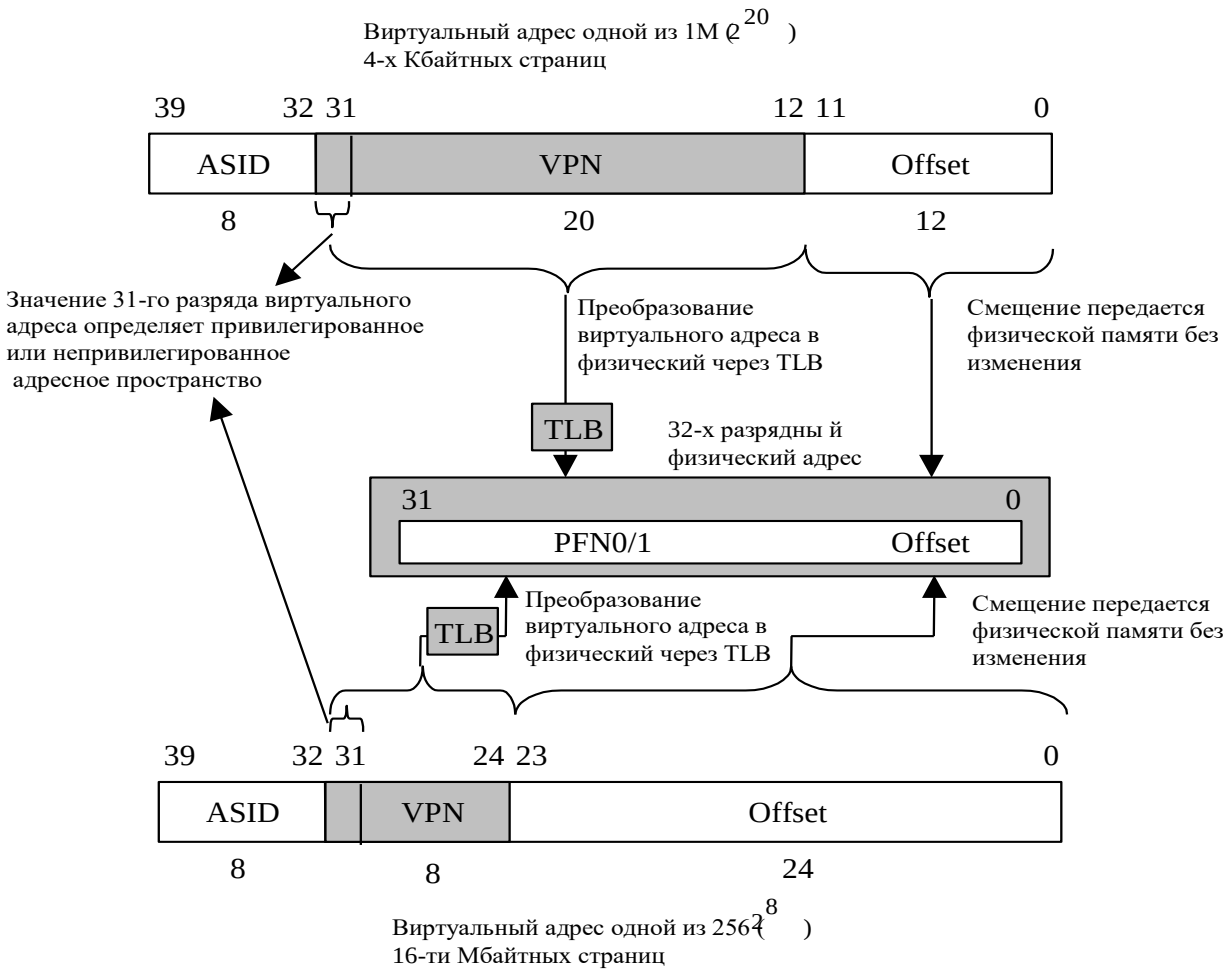


Рисунок 18.15.

18.2.4.8 Попадания (hits), промахи (misses), и множественные попадания (multiple matches)

Каждая строка TLB содержит тэг и два поля данных. Если найдено соответствие, старшие разряды виртуального адреса заменяются физическим номером кадра (PFN), хранящимся в соответствующей строке массива данных TLB. Способ разбиения памяти при отображении определяется в терминах TLB-страниц. TLB поддерживает страницы различных размеров в пределах от 4 Кбайт до 16 Мбайт с шагом по степеням 4. Если соответствие найдено, но строка является инвалидной или запрещенной (т.е., бит V в поле данных равен 0), вырабатывается исключение TLB Invalid.

Если соответствие не найдено, возникает исключение TLB Refill, и программное обеспечение пополняет TLB из таблицы страниц, находящейся в памяти. На Рисунок 18.16 показан алгоритм преобразования и условия возникновения исключений TLB.

Программное обеспечение может делать записи в конкретные строки TLB или использовать аппаратный механизм записи в случайно выбранные строки. Регистр Random определяет, в какую строку будет сделана запись командой TLBWR. Этот регистр декрементируется на каждом такте продвижения конвейера, возвращаясь к максимальному значению после

достижения величины, равной значению регистра Wired. Таким образом, строки TLB, чей номер меньше значения регистра Wired, не затрагиваются командой TLBWR, что позволяет зарезервировать TLB-отображения первостепенной важности.

В режиме TLB также реализован механизм сравнения при записи с целью предотвращения возникновения нескольких соответствий (множественных попаданий). Работает он следующим образом. При выполнении операции записи в TLB, поле VPN2 сравнивается с одноименными полями всех строк TLB. Если будет найдено соответствие, возникнет аппаратно обрабатываемое исключение, которое установит бит TS регистра Status CP0 и прервет эту операцию. Подробно исключения описаны в п. 18.2.5. В каждой строке TLB имеется скрытый бит, обнуляемый при аппаратном сбросе. Устанавливается этот бит при записи в данную строку, разрешая просмотр этой строки при поисках соответствий. Поэтому непроинициализированные строки не вызывают неадекватные преобразования адресов.

Замечание: этот скрытый бит инициализации приводит все строки TLB к инвалидному состоянию после аппаратного сброса, что делает ненужной процедуру очистки (flush) TLB. Но для совместимости с другими MIPS – процессорами рекомендуется заполнять значения тэгов уникальными величинами и обнулять бит валидности (V).

Очистить строку TLB (вывести ее из рассмотрения при поиске) можно, записав в нее значение с неотображаемым через TLB адресом.

Смена размера маски или других переменных строки TLB не приводит к исключению, если она не вводит в противоречие с данной строки другими строками. Например, увеличение размера страницы расширением маски в одной строке TLB может привести к перекрытию данной страницы с другими страницами TLB.

18.2.4.9 Размеры страниц и алгоритм замещения

Для управления общим количеством отображаемого адресного пространства и характеристиками замещения в различных областях памяти ядро обеспечивает два механизма. Первый заключается в том, что размер страницы может быть задан относительно каждой строки TLB, что позволяет отображать страницы размером от 4 Кбайт до 16 Мбайт (по степеням 4). В регистр Page Mask CP0 загружается требуемый размер страницы, который при выполнении операции записи попадает в очередную строку TLB. Таким образом, операционная система может задавать отображения особых назначений. Например, характерный кадровый буфер (frame buffer) может быть отображен на память всего одной строкой TLB.

Второй механизм управляет механизмом замещения, когда возникает промах при просмотре TLB. Для выбора строки TLB, в которую будет записано новое отображение, в процессорном ядре предусмотрен алгоритм случайного замещения. Но существует также способ программно предотвратить случайное замещение зарезервированных отображений, количество которых определяется значением регистра Wired CP0. (см. также п 18.2.7.9).

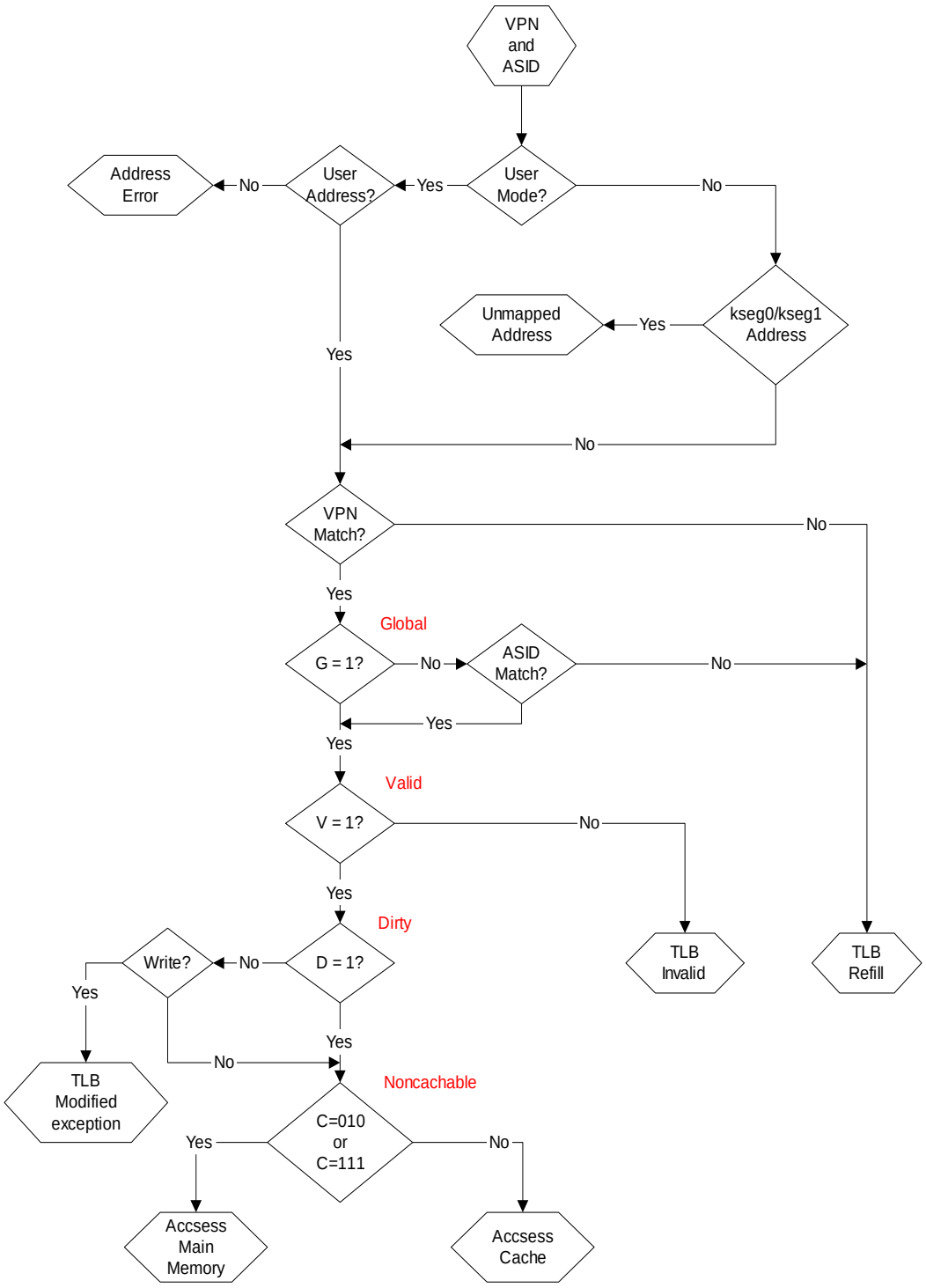


Рисунок 18.16. Алгоритм преобразования адреса через TLB

18.2.5 Исключения

Процессорное ядро способно принимать исключения от ряда источников, в том числе промах буфера преобразования адресов (TLB), арифметическое переполнение, прерывание ввода-вывода, и системные вызовы. Обнаружив одно из этих исключений, CPU

приостанавливает нормальную последовательность исполнения команд и процессор входит в режим Kernel.

В режиме Kernel ядро отключает прерывания и вынуждает процессор запустить программу обработчика исключений, расположенную в фиксированных адресах памяти. Обработчик сохраняет контекст процессора – содержимое счетчика команд, текущий режим процессора и статус разрешения прерываний. Таким образом, контекст может быть восстановлен по завершению обработки исключения.

При возникновении исключения в регистр Exception Program Counter (EPC) загружается адрес, начиная с которого исполнение комад может возобновиться после завершения обработки исключения. В регистр EPC помещается адрес команды, вызвавшей исключение или, если команда находилась в слоте задержки перехода, адрес команды перехода, предшествующей слоту задержки. Чтобы различить эти ситуации, программное обеспечение должно проанализировать бит BD (branch delay) в регистре Cause CP0

18.2.5.1 Условия исключений

Исключения обрабатываются на стадии M конвейера. Когда исключительная ситуация обнаруживается, команда, находящаяся на стадии M, и все команды, следующие за ней на конвейере, отменяются. Соответственно, все условия остановки конвейера, относящиеся к этой команде, а также условия последующих исключений, которые также могут относится к ней, игнорируются, поскольку обслуживание приостановок для отмененной команды не приносит выигрыша.

Когда условие исключения обнаруживается на стадии M, процессор заполняет необходимые регистры CP0 значениями, относящимися к состоянию исключения, изменяет счетчик команд (PC) на адрес соответствующего вектора обработки исключения и очищает признаки исключения, относящиеся к более ранним стадиям конвейера.

Такая реализация позволяет завершить исполнение команды, находящейся на стадии W, и запретить завершение последующих команд. Таким образом, значения, сохраненного в регистре EPC (в случае ошибок – в Error PC), достаточно для возобновления исполнения. Это также обеспечивает поступление исключений в соответствии с порядком исполненя команд – команда, вызывающая исключение, может быть уничтожена командой с более поздней стадии конвейера, также вызвавшей исключение.

18.2.5.2 Приоритеты исключений

В Таблица 18.7 перечислены все возможные исключения со своими относительными приоритетами, от высшего к низшему. Некоторые из этих исключений могут случаться одновременно – в этом случае вызывается исключение с наивысшим приоритетом.

Таблица 18.7.

Исключение	Описание
Reset	Аппаратный сброс
NMI	Внешнее немаскируемое прерывание
TLB_Ri, TLB_Ii	Промах TLB при выборке команды, Попадание в инвалидную страницу TLB (V=0) при выборке команды
AdELi	Ошибка выравнивания адреса при выборке команды; Ссылка на адрес режима Kernel при работе в режиме User при выборке команды
MCheck Sys Bp CpU RI Ov Tr AdELd AdES	Запись в TLB, создающая конфликт с существующей строкой TLB Выполнение команды SYSCALL Выполнение команды BREAK Выполнение команды сопроцессора в режиме User Выполнение зарезервированной команды Переполнение в арифметической команде Выполнение trap (когда условие trap истинно) Ошибка выравнивания адреса при загрузке данных; Ссылка на адрес режима Kernel при работе в режиме User при загрузке данных Ошибка выравнивания адреса при сохранении данных; Попытка сохранения по адресу Kernel в режиме User
TLB_Rd, TLB_Id	Промах TLB при загрузке данных; Попадание в инвалидную страницу TLB (V=0) при загрузке данных
TLB_M	Сохранение в TLB-странице с D=0
Interrupt	Установка незамаскированных HW или SW - прерываний

18.2.5.3 Расположение векторов исключений

Векторы исключений аппаратного сброса и NMI всегда находятся по адресу 0xBFC_0000. Адреса всех других исключений являются комбинациями векторных смещений и базового адреса. В Таблица 18.8 приведены базовые адреса как функции исключения, состояния бита BEV Регистра Status и состояния бита TR_CRAM системного регистра CSR. В Таблица 18.9 приведены смещения от базового адреса как функции исключения. В Таблица 18.10 эти две таблицы сведены в одну таблицу, содержащую все возможные адреса векторов исключений как функции состояний, влияющих на выбор этих векторов.

Таблица 18.8.

<table>
<tr><td rowspan="4">Исключение</td><td colspan="3">$Status_{BEV}$</td></tr>
<tr><td colspan="2">0</td><td rowspan="3">1</td></tr>
<tr><td colspan="2">CSR_{TR_CRAM}</td></tr>
<tr><td>0</td><td>1</td></tr>
<tr><td>Reset, NMI</td><td colspan="3">0xBFC0_0000</td></tr>
<tr><td>Остальные исключения</td><td>0x8000_0000</td><td>0xBFA0_0000 для Risc0
0xA3B0_0000 для Risc1</td><td>0xBFC0_0200</td></tr>
</table>

Таблица 18.9. Базовые адреса векторов исключений

Исключение	Смещение вектора
TLB Refill, EXL = 0	0x000
Reset, NMI	0x00
Исключения общего характера (General Exeptions)	0x180
Interrupt, $Cause_{IV}$ = 1	0x200

Таблица 18.10. Векторы исключений

Исключение	BEV	EXL	IV	CSR_{TR_CRAM}	Вектор
Reset, NMI	–	–	–	–	0xBFC0_0000
TLB Refill	0	0	–	0	0x8000_0000
TLB Refill	0	1	–	0	0x8000_0180
TLB Refill	0	0	–	1	0xBFA0_0000 для Risc0 0xA3B0_0000 для Risc1
TLB Refill	0	1	–	1	0xBFA0_0180 для Risc0 0xA3B0_0180 для Risc1
TLB Refill	1	0	–	–	0xBFC0_0200
TLB Refill	1	1	–	–	0xBFC0_0380
Interrupt	0	0	0	0	0x8000_0180
Interrupt	0	0	1	0	0x8000_0200
Interrupt	0	0	0	1	0xBFA0_0180 для Risc0 0xA3B0_0180 для Risc1
Interrupt	0	0	1	1	0xBFA0_0200 для Risc0 0xA3B0_0200 для Risc1
Interrupt	1	0	0	–	0xBFC0_0380
Interrupt	1	0	1	–	0xBFC0_0400
Остальные	0	–	–	0	0x8000_0180
Остальные	0	–	–	1	0xBFA0_0180 для Risc0 0xA3B0_0180 для Risc1
Остальные	1	–	–	–	0xBFC0_0380

18.2.5.4 Обработка общих исключений

Кроме исключений аппаратного сброса и NMI, которые обслуживаются особым образом, обработка всех остальных исключений происходит в соответствии со следующим основным маршрутом:

- Если бит EXL Регистра Состояния (Status) очищен, в регистр EPC загружается значение PC, по которому выполнение программы будет перезапущено, и при необходимости устанавливается бит BD в Регистре Причины (Cause). Если команда не находится в слоте задержки перехода, бит BD в Регистре Причины будет очищен, а в регистр EPC загружается значение, соответствующее текущему PC. Если же команда находится в слоте задержки перехода, бит BD в Регистре Причины устанавливается в “1”, и в EPC загружается значение, равное PC - 4. Если бит EXL в Регистре Состояния установлен, в регистр EPC ничего не загружается, и бит BD в Регистре Причины не модифицируется.
- В поля CE и ExcCode Регистра Причины загружаются значения, соответствующие исключению.
- Устанавливается бит EXL в Регистре Состояния (Status).
- Процессор стартует с вектора исключения.

Значение, загруженное в EPC, представляет собой адрес возврата из исключения и в обычной ситуации программе обработки исключения не требуется его модифицировать. Программе также не нужно просматривать бит BD в Регистре Причины, если не возникает потребность определить действительный адрес команды, вызвавшей исключение.

```
Operation:

if StatusEXL == 0 then

if InstructionInBranchDelaySlot then

EPC <= PC - 4

CauseBD <= 1

else

EPC <= PC

CauseBD <= 0

endif

if (ExceptionType == TLBRefill) then

vectorOffset <= 0x00

elseif (ExceptionType == Interrupt) and

(CauseIV == 1) then

vectorOffset <= 0x200

else

vectorOffset <= 0x180

endif

else

vectorOffset <= 0x180

endif

CauseCE <= FaultingCoprocessorNumber

CauseExcCode <= ExceptionType

StatusEXL <= 1

if (StatusBEV == 1) then

PC <= 0xBFC0_0200 + vectorOffset
```

else

PC <= 0x8000_0000 + vectorOffset

endif

18.2.5.5 Исключения

В следующих разделах описаны все исключения в порядке, соответствующем Таблице 4.1.

18.2.5.6 Исключение по аппаратному сбросу (Reset Exception)

Это немаскируемое исключение, которое происходит при установке сигнала аппаратного сброса. Когда возникает исключение аппаратного сброса, процессор выполняет полную начальную инициализацию – приводит автоматы к начальному состоянию и переводит процессор в состояние, из которого он может начать запуск команд, находящихся в некэшируемой и неотображаемой области. После возникновения исключения аппаратного сброса состояние процессора не определено, за исключением следующего:

- Регистр Random устанавливается в значение, равное количеству строк TLB - 1.
- Регистр Wired устанавливается в 0.
- Регистр Config устанавливается в свое начальное состояние (boot state).
- Поля RP, BEV, TS, NMI и ERL Регистра Status устанавливаются в заданные значения.
- В PC загружается значение 0xBFC0_0000.

Вектор исключения:

Reset (0xBFC0_0000)

Operation:

Random <= TLBEntries – 1

Wired <= 0

Config <= ConfigurationState

$Status_{RP}$ <= 0

$Status_{BEV}$ <= 1

$Status_{TS}$ <= 0

$Status_{NMI}$ <= 0

$Status_{ERL}$ <= 1

PC <= 0xBFC0_0000

18.2.5.7 Исключение по немаскируемому прерыванию (Non Maskable Interrupt – NMI Exception)

Немаскируемое прерывание возникает по положительному фронту входного сигнала NMI или при срабатывании сторожевого таймера WDT. Исключение NMI происходит только в пределах границ команды, поэтому оно не вызывает сброса или другую переинициализацию аппаратных средств. Состояние кэш, памяти, а также другие состояния процессора остаются неизменными. Значения регистров также сохраняются за исключением следующего:

- Поля BEV, TS, NMI и ERL регистра Status принимают заданные значения.
- В регистр ErrorEPC загружается значение PC - 4, если прерывание произошло на фоне команды в слоте задержки перехода. В противном случае в регистр ErrorEPC загружается значение PC.
- В PC загружается значение 0xBFC0_0000.

Вектор исключения:

Reset (0xBFC0_0000)

Operation:

StatusBEV <= 1

StatusTS <= 0

StatusNMI <= 1

StatusERL <= 1

if InstructionInBranchDelaySlot then

ErrorEPC <= PC - 4

else

ErrorEPC <= PC

endif

PC <= 0xBFC0_0000

18.2.5.8 Исключение по обновлению TLB — выборка команды или доступ к данным (TLB Refill Exception – Instruction Fetch or Data Access)

Исключение TLB Refill происходит во время выборки команды или доступа к данным, если в TLB нет ни одной строки, соответствующей ссылке к отображенному адресному пространству, и бит EXL в регистре Status равен 0.

Значение поля ExcCode регистра Cause:

TLBL: Произошла ссылка по загрузке данных или выборке команды

TLBS: Произошла ссылка по сохранению данных

Дополнительно сохраняемые состояния:

Таблица 18.11.

Состояние регистра	Значение
BadVAddr	ошибочный адрес
Context	поле BadVPN2 содержит $VA_{31:13}$ ошибочного адреса
EntryHi	поле VPN2 содержит $VA_{31:13}$ ошибочного адреса; поле ASID содержит ASID отсутствующей ссылки

Вектор исключения:

Вектор TLB Refill (смещение 0x00)

18.2.5.9 Исключение по инвалидности TLB — выборка команды или доступ к данным (TLB Invalid Exception – Instruction Fetch or Data Access)

Исключение TLB Invalid происходит во время выборки команды или доступа к данным в одном из следующих случаев:

- В TLB нет ни одной строки, соответствующей ссылке к отображенному адресному пространству, и бит EXL в регистре Status равен 1.
- Строка TLB соответствует ссылке к отображенному адресу, но ее бит валидности выключен.

Значение поля ExcCode регистра Cause:

TLBL: Произошла ссылка по загрузке данных или выборке команды

TLBS: Произошла ссылка по сохранению данных

Дополнительно сохраняемые состояния:

Таблица 18.12.

Состояние регистра	Значение
BadVAddr	ошибочный адрес
Context	поле BadVPN2 содержит $VA_{31:13}$ ошибочного адреса
EntryHi	поле VPN2 содержит $VA_{31:13}$ ошибочного адреса; поле ASID содержит ASID отсутствующей ссылки

Вектор исключения:

Общий Вектор исключения (смещение 0x180)

18.2.5.10 Исключение по ошибке адресации— выборка команды / доступ к данным (Address Error Exception – Instruction Fetch / Data Access)

Исключение по ошибке адресации во время доступа к команде или данным возникает при попытке выполнить одно из следующих действий:

- Выбрать команду, загрузить или сохранить слово данных, если они не выравнены в границах слова
- Загрузить или сохранить пол-слова, если оно не выравнено в границах пол-слова
- Обратиться по адресу пространства Kernel при работе в режиме User

Значение поля ExcCode регистра Cause:

ADEL: Произошла ссылка по загрузке данных или выборке команды

ADES: Произошла ссылка по сохранению данных

Дополнительно сохраняемые состояния:

Таблица 18.13.

Состояние регистра	Значение
BadVAddr	ошибочный адрес

Вектор исключения:

Общий Вектор исключения (смещение 0x180)

18.2.5.11 Исключение по аппаратному контролю (Mcheck – Machine Check Exception)

Данное исключение возникает, если при выполнении команды записи в TLB (TLBWI или TLBWR) обнаруживается, что поле виртуального адреса записываемой строки соответствует такому же полю одной из строк, уже хранящихся в TLB.

При возникновении данной ситуации запись в TLB не выполняется и устанавливается бит TS в регистре Status. Этот бит является статусным и не влияет на функционирование процессорного ядра. Сбрасывается он программно после разрешения данной ситуации, осуществляемого очисткой конфликтных строк в TLB.

Значение поля ExcCode регистра Cause:

Mcheck

Дополнительно сохраняемые состояния:

Нет

Вектор исключения:

Общий Вектор исключения (смещение 0x180)

18.2.5.12 Исключение исполнения – системный вызов (System Call Exception)

Исключение System Call является одним из шести исключений исполнения. Все такие исключения имеют одинаковый приоритет. Исключение System Call возникает при исполнении команды SYSCALL.

Значение поля ExcCode регистра Cause:

Sys

Дополнительно сохраняемые состояния:

Нет

Вектор исключения:

Общий Вектор исключения (смещение 0x180)

18.2.5.13 Исключение исполнения — Breakpoint (Execution Exception – Breakpoint)

Исключение Breakpoint является одним из шести исключений исполнения. Все такие исключения имеют одинаковый приоритет. Исключение Breakpoint возникает при исполнении команды BREAK.

Значение поля ExcCode регистра Cause:

Bp

Дополнительно сохраняемые состояния:

Нет

Вектор исключения:

Общий Вектор исключения (смещение 0x180)

18.2.5.14 Исключение исполнения — зарезервированная команда (Execution Exception – Reserved Instruction)

Исключение зарезервированной команды является одним из шести исключений исполнения. Все такие исключения имеют одинаковый приоритет. Исключение зарезервированной команды вызывается при исполнении команды с неопределенным старшим кодом операции (major opcode) или полем функции.

Значение поля ExcCode регистра Cause:

RI

Дополнительно сохраняемые состояния:

Нет

Вектор исключения:

Общий Вектор исключения (смещение 0x180)

18.2.5.15 Исключение исполнения — недоступен сопроцессор (Execution Exception – Coprocessor Unusable)

Исключение недоступности сопроцессора является одним из шести исключений исполнения. Все такие исключения имеют одинаковый приоритет. Исключение недоступности сопроцессора вызывается при попытке исполнения команды сопроцессора CP0 в режиме User, если сопроцессор не был заказан для использования.

Значение поля ExcCode регистра Cause:

CpU

Дополнительно сохраняемые состояния:

Нет

Вектор исключения:

Общий Вектор исключения (смещение 0x180)

18.2.5.16 Исключение исполнения — целочисленное переполнение (Execution Exception – Integer Overflow)

Исключение целочисленного переполнения является одним из шести исключений исполнения. Все такие исключения имеют одинаковый приоритет. Исключение целочисленного переполнения вызывается, когда выбранные целочисленные команды приводят к переполнению в двоичном коде.

Значение поля ExcCode регистра Cause:

Ov

Дополнительно сохраняемые состояния:

Нет

Вектор исключения:

Общий Вектор исключения (смещение 0x180)

18.2.5.17 Исключение исполнения — Trap (Execution Exception – Trap)

Исключение Trap является одним из шести исключений исполнения. Все такие исключения имеют одинаковый приоритет. Исключение Trap вызывается, если условие команды trap истинно (TRUE).

Значение поля ExcCode регистра Cause:

Tr

Дополнительно сохраняемые состояния:

Нет

Вектор исключения:

Общий Вектор исключения (смещение 0x180)

18.2.5.18 Исключение сохранения в запрещенной области (TLB Modified Exception)

Это исключение возникает при обращении по записи данных к отображенному адресу, если выполняется следующее условие:

- Найденная строка TLB действительна, но страница запрещена для записи.

Значение поля ExcCode регистра Cause:

Mod

Дополнительно сохраняемые состояния:

Таблица 18.14.

Состояние регистра	Значение
BadVAddr	Ошибочный адрес
Context	Поля BadVPN2 содержат $VA_{31:13}$ ошибочного адреса
EntryHi	Поле VPN2 содержит $VA_{31:13}$ ошибочного адреса; поле ASID содержит ASID отсутствующей ссылки

Вектор исключения:

Общий Вектор исключения (смещение 0x180)

18.2.5.19 Исключение прерывания (Interrupt Exception)

Исключение прерывания возникает, когда сигнал одного или более разрешенных регистром Status прерываний устанавливается на входе процессора.

Значение поля ExcCode регистра Cause:

Mod

Дополнительно сохраняемые состояния:

Таблица 18.15.

Состояние регистра	Значение
$Cause_{IP}$	Указывает код прерывания

Вектор исключения:

Общий Вектор исключения (смещение 0x180), если бит IV регистра Cause равен 0;

Вектор прерывания (смещение 0x200), если бит IV регистра Cause равен 1.

18.2.6 Алгоритмы обработки исключений

В этом разделе приведены алгоритмы обработки следующих исключений:

- Общие исключения;
- Исключения пропуска при поиске по TLB;
- Исключения Reset и NMI;

Исключения аппаратно обрабатываются, а затем программно обслуживаются.

Алгоритмы обработки исключений приведены на Рисунок 18.17, Рисунок 18.18, Рисунок 18.19.

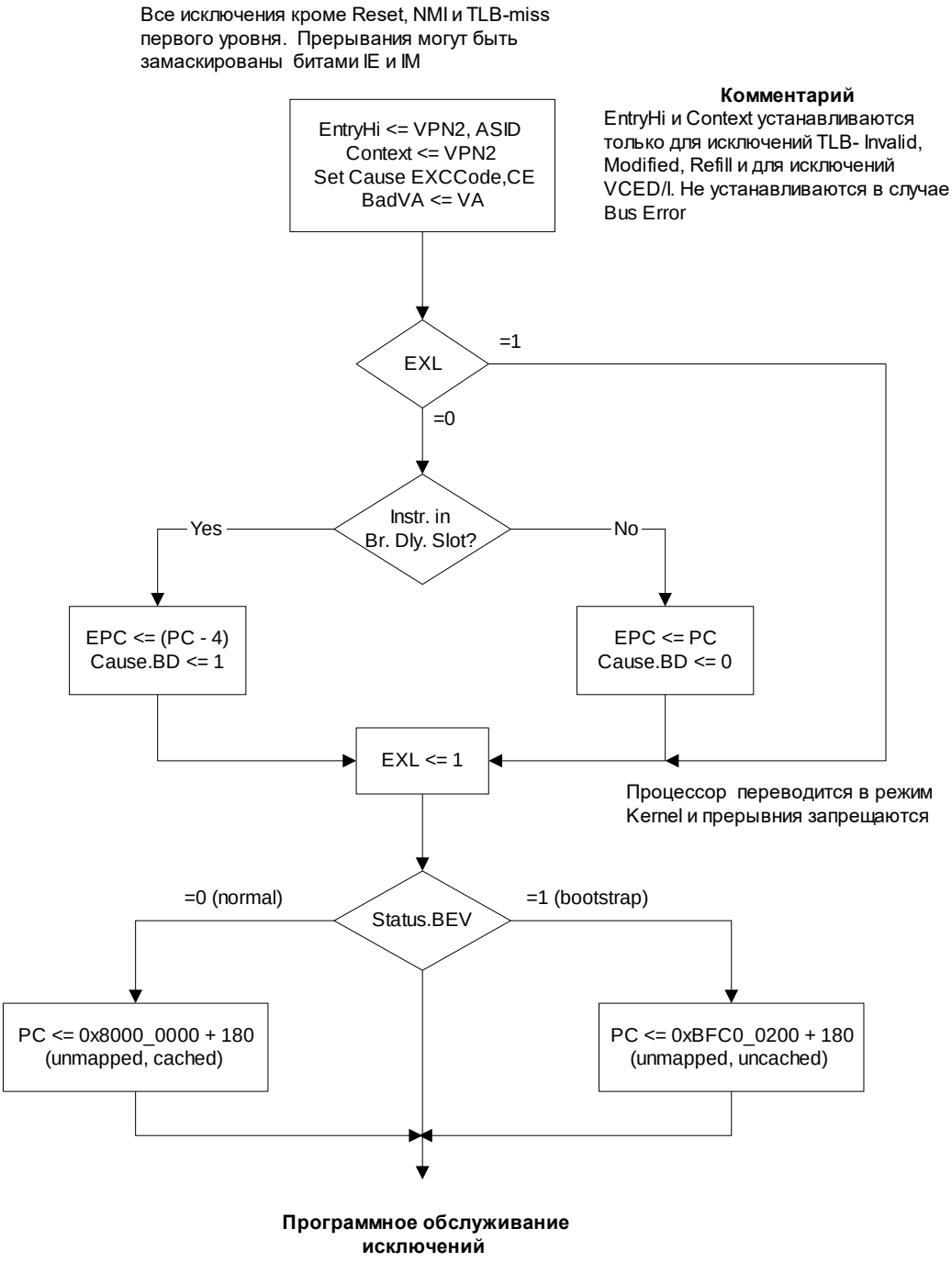


Рисунок 18.17. Обработка общих исключений

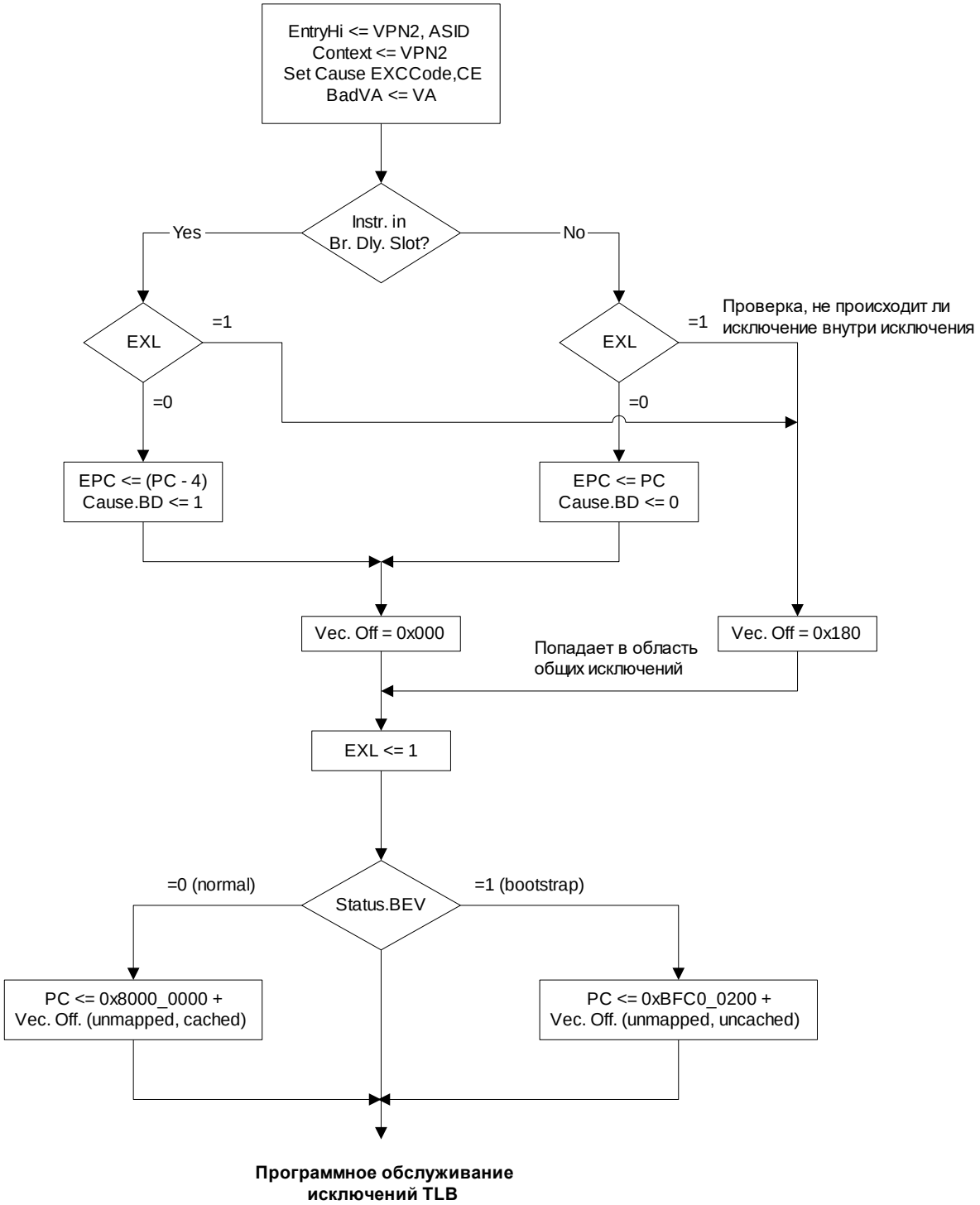


Рисунок 18.18. Обработка исключений TLB Refill и TLB Invalid

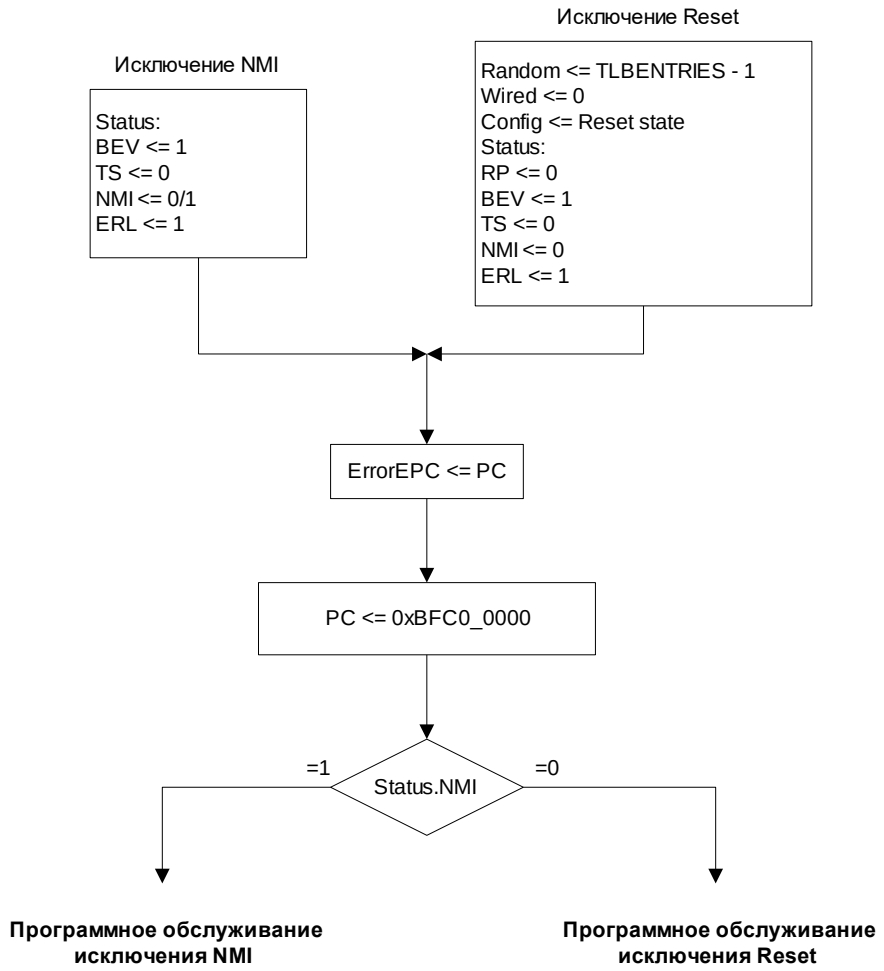


Рисунок 18.19. Обработка исключений Reset и NMI

18.2.7 Регистры CP0

18.2.7.1 Назначение

Системный Управляющий Сопроцессор (CP0) обеспечивает регистровый интерфейс с процессорным ядром MIPS32 и поддерживает управление памятью, преобразование адреса, обработку исключений и другие привилегированные операции. Каждому регистру CP0 соответствует определяющий его уникальный номер; этот номер называется *номером регистра*. Например, регистру PageMask соответствует 5-й номер регистра.

После записи нового значения в регистр CP0 (с помощью команды MTC0), его обновление происходит не сразу, а по прошествии периода от 0 и более команд. Этот период называется периодом особой ситуации (см. Раздел 2.6).

18.2.7.2 Обзор регистров CP0

В Таблица 18.16 приведены все регистры CP0 в порядке возрастания нумерации. В Разделе 5.3 каждый из этих регистров описан отдельно.

Таблица 18.16. Регистры CP0

Номер регистра	Название Регистра	Функция
0	Index[1]	Индекс матрицы TLB (режим TLB)
1	Random[1]	Случайным образом сгенерированный индекс для буфера TLB (режим TLB)
2	EntryLo0[1]	Младшая часть строки TLB для виртуальных страниц с четными номерами (режим TLB)
3	EntryLo1[1]	Младшая часть строки TLB для виртуальных страниц с нечетными номерами (режим TLB)
4	Context[2]	Указатель на строку в таблице страниц памяти (режим TLB)
5	PageMask[1]	Управление переменным размером страниц строк TLB (режим TLB)
6	Wired[1]	Управление количеством закрепленных “привязанных” строк TLB (режим TLB)
7	–	Резерв
8	BadVAddr[2]	Содержит адрес, вызвавший последнее связанное с адресацией исключение
9	Count[2]	Счетчик процессорных циклов
10	EntryHi[1]	Старшая часть строки TLB (режим TLB)
11	Compare[2]	Управление прерыванием таймера
12	Status[2]	Состояние и управление процессором
13	Cause[2]	Причина последнего исключения
14	EPC[2]	Значение счетчика команд во время последнего исключения
15	PRId	Идентификация и ревизия процессора
16	Config/Config1	Конфигурационный регистр
17	LLAddr	Загрузка адреса сопряжения
18-19	Не реализованы	
20	PRnumber	Номер процессора в кластере
21-22	–	Резерв
23-24	Не реализованы	
25-27	–	Резерв
28-29	Не реализованы	
30	ErrorEPC[2]	Значение счетчика команд при последней ошибке
31	Не реализован	

[1]Регистры, используемые при управлении памятью.

[2]Регистры, используемые при обработке исключений.

18.2.7.3 Регистры CP0

Регистры CP0 обеспечивают интерфейс между системой команд (ISA) и архитектурой процессора. Каждый регистр, описанный в этом разделе, представлен своим порядковым номером и значением поля select.

Все поля описанных регистров характеризуются свойствами записи / чтения, а также значением после аппаратного сброса. Свойства записи / чтения охарактеризованы в Таблица 18.17.

Таблица 18.17.

Свойства записи/чтения	Аппаратная интерпретация	Программная интерпретация
R/W	Поле, в котором все биты программно и аппаратно доступны по записи и чтению. Аппаратное обновление этого поля доступно для программы при чтении программой. Программное обновление этого поля доступно для процессора при чтении процессором. Если значение поля после сброса не определено, программа или процессор должны проинициализировать это поле, чтобы первое чтение возвратило предсказуемое значение.	
R	Поле, значение которого постоянно или обновляется только процессором. Значение поля после начальной установки восстанавливается также при включении питания. Если значение поля не определено после начальной установки, процессор обновляет его только при условиях, определенных при описании поля.	Поле, для которого значение, записанное программой, процессором игнорируется. Программное прочтение этого поля возвращает последнее обновленное процессором значение. Если значение поля не определено после начальной установки, программное прочтение этого поля возвратит непредсказуемое значение кроме тех случаев, когда произошло обновление процессором значения этого поля по возникновению условий, определенных в описании поля условий.
0	Поле, значение которого процессором не обновляется и всегда равно нулю.	Программное чтение всегда возвращает нуль.

18.2.7.4 Регистр Index (Регистр 0 CP0, Select 0).

Регистр Index является 32-х разрядным регистром, доступным для чтения и записи. Он содержит индекс доступа к TLB для команд TLBP, TLBR и TLBWI. Ширина поля индекса зависит от количества строк TLB и равна 4.

Функционирование процессора НЕОПРЕДЕЛЕНО, если в регистр Index записано значение большее или равное количеству строк TLB.

Формат регистра Index

31	30 ... 4	3 ... 0
P	0	Index

Таблица 18.18. Описание полей регистра Index

Номер бита	Поле	Тип	Описание	Сброс
31	P	R	Неудачная проба. Устанавливается в 1, если предыдущей командой TLBProbe (TLBP) не было найдено соответствия в TLB.	Не определено
30:4	0	0	При чтении возвращается нуль	0
3:0	Index	R/W	Индекс строки TLB, к которой относятся команды TLBRead и TLBWrite	Не определено

18.2.7.5 Регистр Random (Регистр CP0 1, Select 0).

Регистр Random доступен только для чтения, и его значение используется как индекс TLB для команды TLBWR. Ширина поля Random определяется таким же образом, как для регистра Index.

Значение этого регистра изменяется между верхней и нижней границами следующим образом:

- Нижняя граница определяется количеством строк TLB , зарезервированных для использования операционной системой (содержимое регистра Wired). Строка, чей индекс равен значению Wired, является первой из доступных для записи командой TLB Write Random (TLBWR).
- Верхняя граница равна общему количеству строк TLB минус 1.

Регистр Random уменьшается на 1 при продвижении конвейера RISC, возвращаясь к максимальному значению по достижению величины, равной значению регистра Wired.

Процессор инициализирует регистр Random значением, равным верхней границе по возникновению исключения Reset и по записи в регистр Wired.

Формат регистра Random

31	4	3	0
0		Random	

Таблица 18.19. Описание полей регистра Random

Номер бита	Поле	Тип	Описание	Сброс
31:4	0	0	При чтении возвращается нуль	0
3:0	Random	R	Случайный индекс строки TLB	TLB Entries - 1

18.2.7.6 EntryLo0, EntryLo1 (Регистры 2 и 3 CP0, Select 0)

Пара регистров EntryLo действует как интерфейс между TLB и командами TLBR, TLBWI, TLBWR.

В режиме TLB EntryLo0 содержит строки для четных страниц TLB, а EntryLo1 – для нечетных страниц.

После ошибки адресации и возникновения исключений TLB refill, TLB invalid и TLB modified, содержимое регистров EntryLo0 и EntryLo1 не определено.

Формат регистров EntryLo0, EntryLo1

31	30 29	26 25	6 5	3 2	1	0
R	0	PFN	C	D	V	G

Таблица 18.20. Описание полей регистров EntryLo0 и EntryLo1

Номер бита	Поле	Тип	Описание	Сброс
31:30	R	R	Резервные. При чтении возвращается нуль	0
29:26	0	R	При чтении возвращается нуль	0
25:6	PFN	R/W	Номер страничного кадра. Соответствует битам 31:12 физического адреса.	Не определено
5:4	0	R	Не используются. При чтении возвращается нуль.	0
5:3	C	R/W	Атрибут когерентности страницы. См. табл.5.6	Не определено
2	D	R/W	“Dirty” – бит, разрешающий запись. Указывает на то, что в страницу была сделана запись, и/или страница открыта для записи. Если этот бит равен 1, разрешается сохранение в этой странице. Если он равен 0, сохранение в этой странице вызывает исключение TLB Modified.	Не определено
1	V	R/W	Бит валидности. Указывает, на то, что строка TLB и, соответственно, отображение виртуальной страницы, является действительным. Если этот бит равен 1, доступ к странице разрешается. Если этот бит равен 0, доступ к странице вызывает исключение TLB Invalid.	Не определено
0	G	R/W	Бит глобальности. При записи в TLB битом G в строке TLB становится логическое “И“ битов G EntryLo0 и EntryLo1. Если бит G строки TLB равен 1, результат сравнения полей ASID игнорируется при поиске по TLB. При чтении строки TLB биты G EntryLo0 и EntryLo1 отражают состояние бита G TLB.	Не определено

В Таблица 18.21 приведена декодировка для поля C регистров EntryLo0 и EntryLo1 и полей K0, K23 и KU регистра Config.

Таблица 18.21. Атрибуты когерентности Кэш

Значение C[5:3]	Описание
0, 1, 3*, 4, 5, 6	Кэшируемая, некогерентная область
2*, 7	Некэшируемая область
* - Архитектура MIPS32 предусматривает только эти два значения. Остальные значения не используются и отображаются в используемые значения. Например, 0, 1, 4, 5 и 6 отображается в 3, а 7 – в 2.	

18.2.7.7 Регистр Context (Регистр 4 CP0, Select 0)

Регистр Context доступен для чтения и записи, и содержит указатель на строку в матрице PTE (page table entry). Эта матрица является структурой данных операционной системы, в которой содержатся преобразования виртуального адреса в физический. При возникновении промаха TLB, операционная система загружает в TLB недостающее преобразование из матрицы PTE. Регистр Context дублирует часть информации,

содержащейся в регистре BadVAddr, но организован таким образом, что операционная система может прямо ссылаться к 8-байтной матрице PTE в памяти.

При возникновении исключения TLB (TLB Refill, TLB Invalid, или TLB Modified) биты $VA_{31:13}$ виртуального адреса записываются в поле BadVPN2 регистра Context. Поле PTEBase записывается и используется операционной системой.

После возникновения исключения ошибки адресации значение поля BadVPN2 регистра Context не определено.

Формат регистра Context

31	23	22	4	3	0
PTEBase		BadVPN2			

Таблица 18.22. Описание полей регистра Context

Номер бита	Поле	Тип	Описание	Сброс
31:23	PTEBase	R/W	Это поле используется операционной системой и обычно содержит значение, позволяющее операционной системе использовать регистр Context в качестве указателя на текущую матрицу PTE в памяти.	Не определено
22:4	BadVPN2	R	Это поле заполняется процессором при промахе TLB. Оно содержит биты $VA_{31:13}$ пропущенного виртуального адреса	Не определено
3:0	0	0	При чтении возвращается нуль	0

18.2.7.8 Регистр PageMask (Регистр 5 CP0, Select 0)

Регистр PageMask доступен для чтения и записи, и используется для чтения TLB и записи в TLB. Он содержит маску сравнения, которая устанавливает переменную размера страниц для каждой строки TLB, как показано в Таблица 18.24. Если значение регистра отлично от значений, приведенных в таблице, поведение процессора при поиске по TLB не определено.

Формат регистра PageMask

31	25	24	13	12	0
0		Mask		0	

Таблица 18.23. Описание полей регистра PageMask

Номер бита	Поле	Тип	Описание	Сброс
24:13	Mask	R/W	Бит маски, содержащий “1”, указывает на то, что соответствующий бит виртуального адреса не должен принимать участие при поиске соответствия по TLB	Не определено
31:25, 12:0	0	0	При чтении возвращается нуль	0

Таблица 18.24. Таблица возможных значений поля Mask регистра PageMask.

Размер страницы	Бит											
	24	23	22	21	20	19	18	17	16	15	14	13
4 КБ	0	0	0	0	0	0	0	0	0	0	0	0
16 КБ	0	0	0	0	0	0	0	0	0	0	1	1

64 КБ	0	0	0	0	0	0	0	0	1	1	1	1
256 КБ	0	0	0	0	0	0	1	1	1	1	1	1
1 МБ	0	0	0	0	1	1	1	1	1	1	1	1
4 МБ	0	0	1	1	1	1	1	1	1	1	1	1
16 МБ	1	1	1	1	1	1	1	1	1	1	1	1

18.2.7.9 Регистр Wired (Регистр 6 CP0, Select 0)

Регистр Wired доступен для чтения и записи. Этот регистр определяет границу между случайными и “привязанными” строками TLB, как показано на Рисунок 18.20. Ширина поля Wired определяется так же, как для описанного выше регистра Index. “Привязанные” строки зафиксированы – они не являются удаляемыми и не могут быть перезаписаны командой TLBWR. Эти строки могут быть перезаписаны только командой TLBWI.

Регистр Wired устанавливается в нулевое состояние исключением по аппаратному сбросу (Reset). Запись в регистр Wired вызывает установку регистра Random в значение, равное его верхней границе.

Если значение, записанное в регистр Wired, больше или равно числу строк TLB, операция процессора не определена.

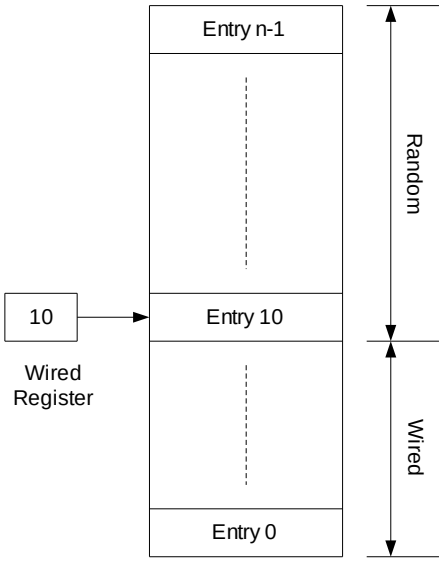


Рисунок 18.20. “Привязанные” и случайные строки TLB

Формат регистра Wired

31	4	3	0
0		Wired	

Таблица 18.25. Описание полей регистра Wired

Номер бита	Поле	Тип	Описание	Сброс
31:4	0	0	При чтении возвращается нуль	0
3:0	Wired	R/W	Граница между “привязанными” и случайными строками TLB.	0

18.2.7.10 Регистр BadVAddr (Регистр 8 CP0, Select 0)

Регистр BadVAddr доступен только для чтения и содержит последний виртуальный адрес, вызвавший одно из следующих исключений:

- Ошибка адреса (AdEL или AdES)
- TLB Refill
- TLB Invalid
- TLB Modified

Формат регистра BadVAddr

31	0
BadVAddr	

Таблица 18.26. Описание полей регистра BadVAddr

Номер бита	Поле	Тип	Описание	Сброс
31:0	BadVAddr	R	Виртуальный адрес исключения	Не определено

18.2.7.11 Регистр Count (Регистр 9 CP0, Select 0)

Регистр Count действует как таймер, увеличивающий свое значение каждый такт.

Регистр Count может быть записан в функциональных или диагностических целях, включая установку или синхронизацию процессора.

Формат регистра Count

31	0
Count	

Таблица 18.27. Описание полей регистра Count

Номер бита	Поле	Тип	Описание	Сброс
31:0	Count	R/W	Счетчик	Не определено

18.2.7.12 Регистр EntryHi (Регистр 10 CP0, Select 0)

Регистр EntryHi содержит информацию соответствия виртуального адреса, использующуюся при чтении, записи и операциях доступа к TLB.

При возникновении исключений TLB (TLB Refill, TLB Invalid или TLB Modified) биты $VA_{31:13}$ виртуального адреса записываются в поле VPN2 регистра EntryHi. В поле ASID, которое используется в процессе сравнения при поиске по TLB, программно записывается идентификатор текущего адресного пространства.

Поле VPN2 регистра EntryHi не определено после прерывания по ошибке адресации.

Формат регистра EntryHi

31		0
VPN2	0	ASID

Таблица 18.28. Описание полей регистра EntryHi

Номер бита	Поле	Тип	Описание	Сброс
31:13	VPN2	R/W	Разряды $VA_{31:0}$ виртуальнго адреса (виртуальный номер страницы, деленный на 2). Это поле записывается аппаратно при исключении TLB или при чтении TLB, и программно перед записью в TLB.	Не определено
12:8	0	0	При чтении возвращается нуль	0
7:0	ASID	R/W	Идентификатор адресного пространства. Это поле записывается аппаратно при чтении TLB, и программно при установке текущего значения ASID для записи в TLB и для сравнения при поиске по TLB с соответствующими полями ASID в строках TLB.	Не определено

18.2.7.13 Регистр Compare (Регистр 11 CP0, Select 0)

Регистр Compare действует совместно с регистром Count с целью реализации функции таймера и прерывания по таймеру. Прерывание по таймеру является выходным сигналом процессора.

Когда значение регистра Count равняется значению регистра Compare, выход процессора SI_TimerInt устанавливается в 1. Этот выход остается равным 1, пока в регистр Compare не будет произведена запись. SI_TimerInt может быть заведен обратно в процессор на один из входов аппаратных прерываний. Обычно это делается мультиплексированием его с аппаратным прерыванием 5 для установки бита прерывания IP(7) в регистре Cause.

Для диагностических целей регистр Compare доступен для чтения и записи. Однако при нормальном функционировании регистр Compare используется только для записи. При записи значения в регистр Compare в качестве побочного эффекта происходит очистка прерывания по таймеру.

Формат регистра Compare

31	0
Compare	

Таблица 18.29. Описание полей регистра Compare

Номер бита	Поле	Тип	Описание	Сброс
31:0	Compare	R/W	Период счета таймера	Не определено

18.2.7.14 Регистр Status (Регистр 12 CP0, Select 0)

Регистр Status (SR) является регистром, доступным для чтения и записи. Он содержит поля рабочего режима, разрешения прерываний и диагностические состояния процессора. Для задания режимов функционирования процессора поля этого регистра объединяются следующим образом:

Разрешение прерываний: Прерывания разрешаются, когда истинны все следующие условия:

- IE = 1
- EXL = 0
- ERL = 0

Если эти условия выполнены, прерывания разрешаются установкой битов IM.

Рабочие режимы: Процессор всегда находится в одном из двух режимов – Kernel или User. Режим задается установкой следующих битов регистра Status CPU.

- Режим User: UM = 1, EXL = 0, and ERL = 0
- Режим Kernel: UM = 0 или EXL = 1 или ERL = 1

Формат Status регистра

31 28	27	26 23	22	21	20	19	18 8	7 5	4	3	2	1	0
CU3-CU0	RP	0	BEV	TS	0	NMI	IM10-IM0	0	UM	0	ERL	EXL	IE

Таблица 18.30. Описание полей регистра Status

Номер бита	Поле	Тип	Описание	Сброс
31:28	CU3-CU0	R/W	Не используются	Не определено
27	RP	R/W	Разрешает режим пониженного потребления. Состояние RP бита выдается наружу как сигнал SI_RP.	0
26:23	-	0	При чтении возвращается нуль	0
22	BEV	R/W	Управление размещением векторов исключения: 0: Нормальный 1: Начальная загрузка	1
21	TS	R/W	TLB-закрытие системы. Этот бит устанавливается, если при выполнении команд TLBWI или TLBWR образуется команда, которая приводит к условию закрытия, если оно разрешено. Программа может записывать в этот разряд только 0 чтобы очистить его, и не может вызвать переход этого бита из 0 в 1.	0
19	NMI	R/W	Указывает, что вход в вектор исключения начальной установки был осуществлен по причине возникновения NMI. 0: Не NMI (Аппаратный сброс) 1: NMI Программное обеспечение может записывать в этот бит только 0, чтобы очистить его, и не может записать 1.	1 для NMI, иначе 0
18:8	IM[10:0]	R/W	Маска Прерывания: Управление разрешением внешних, внутренних и программных прерываний. Прерывание принимается в случае, если прерывания разрешены и установлены соответствующие биты как в поле Маски Прерывания (IM) регистра Status, так и в поле установки прерываний регистра Cause, а также установлен бит IE регистра Status. 0: Запрос на прерывание не разрешен 1: Запрос на прерывание разрешен	Не определено

Номер бита	Поле	Тип	Описание	Сброс
7:5	-	0	При чтении возвращается нуль	0
4	UM	R/W	Указывает на то, что процессор работает в непривилегированном режиме (User): 0: Процессор работает в привилегированном режиме (Kernel) 1: Процессор работает в непривилегированном режиме (User) Замечание: процессор может также находиться в режиме Kernel, если установлены биты EXL или ERL. Это условие не влияет на состояние бита UM.	Не определено
3	-	0	При чтении возвращается нуль	0
2	ERL	R/W	Уровень ошибки. Устанавливается процессором при возникновении исключений Reset и NMI. 0: Нормальный уровень 1: Уровень ошибки Когда бит ERL установлен: Процессор находится в режиме Kernel. Прерывания запрещены. Команда ERET использует адрес возврата, содержащийся в ErrorEPC вместо EPC. kuseg используется как неотображаемая и некэшируемая область. Это позволяет иметь доступ к главной памяти при ошибках кэш. Поведение процессора не определено если бит ERL установлен при выполнении кода из useg/kuseg.	1
1	EXL	R/W	Уровень Исключения. Устанавливается процессором при возникновении любого исключения, кроме Reset и NMI. 0: Нормальный уровень 1: Уровень исключения Когда бит EXL установлен: Процессор переходит в привилегированный режим (Kernel). Прерывания запрещены. Исключения TLB Refill используют общий вектор исключения вместо вектора TLB Refill. Если происходит другое исключение, EPC не модифицируется.	Не определено
0	IE	R/W	Разрешение Прерывания. 0: Отключает прерывания 1: Разрешает прерываниям	Не определено

18.2.7.15 Регистр Cause (Регистр 13 CP0, Select 0)

Регистр Cause, в основном, описывает причину последнего исключения. Кроме того, поля регистра управляют запросами на программные прерывания и определяют вектор, которым обрабатываются прерывания. Все поля регистра Cause, за исключением IP[1:0], IV и WP, доступны только для чтения.

Формат регистра Cause

31	30 24	23	22 19	18 10	9 8	7	6 2	1 0
BD	0	IV	0	IP[10:2]	IP[1:0]	0	Exc Code	0

Таблица 18.31. Описание полей регистра Cause

Номер бита	Поле	Тип	Описание	Сброс
31	BD	R	Указывает на то, что последнее исключение произошло в слоте задержки перехода: 0: Не в слоте задержки 1: В слоте задержки Замечание: бит BD не модифицируется на новом исключении, если установлен бит EXL.	Не определено
30:24	0	0	При чтении возвращается нуль	0
23	IV	R/W	Указывает, какой вектор используется для обслуживания исключений прерывания – общий или специальный вектор прерываний: 0: Используется общий вектор исключения (0x180) 1: Используется специальный вектор прерываний (0x200)	Не определено
22:16	0	0	При чтении возвращается нуль	0
15:10	IP[7:2]	R	Указывает, какое внешние прерывание установлено: 15: Резерв 14: Прерывание по Count/Compare 13: • для Risc0 Прерывание QLIC0 TARG3 • для Risc1 Прерывание QLIC1 TARG3 12: • для Risc0 Прерывание QLIC0 TARG2 • для Risc1 Прерывание QLIC1 TARG2 11: • для Risc0 Прерывание QLIC0 TARG1 • для Risc1 Прерывание QLIC1 TARG1 10: • для Risc0 Прерывание QLIC0 TARG0 • для Risc1 Прерывание QLIC1 TARG0	Не определено
9:8	IP[1:0]	R/W	Управляет запросом программных прерываний: 9: Запрос программного прерывания 1 8: Запрос программного прерывания 0	Не определено
7	0	0	При чтении возвращается нуль	0
6:2	Exc Code		Код исключения — см. Таблица 18.32	
1:0	0	0	При чтении возвращается нуль	0

Таблица 18.32. Описание поля Exc Code регистра Cause

Значение Exc Code	Мнемоника	Описание
0	Int	Прерывание
1	Mod	TLB-исключение модификации
2	TLBL	TLB-исключение (загрузка или вызов команды)
3	TLBS	TLB-исключение (сохранение)
4	AdEL	Прерывание по ошибке адресации (загрузка или вызов команды)
5	AdES	Прерывание по ошибке адресации (сохранение)
6-7		Не используются
8	Sys	Системное исключение
9	Bp	Исключение Breakpoint
10	RI	Исключение зарезервированной команды
11	CpU	Исключение недоступности сопроцессора
12	Ov	Исключение целочисленного переполнения
13	Tr	Исключение Trap
14-22	–	Зарезервированы
23		Не используется

24	MCheck	Аппаратный контроль
25-31	–	Зарезервированы

18.2.7.16 Регистр EPC (Регистр 14 CP0, Select 0)

Программный счетчик исключения (EPC) является регистром, доступным для чтения и записи. EPC содержит адрес, начиная с которого возобновляется исполнение программы после завершения обработки исключения. Все биты регистра EPC значимы и должны перезаписываться.

Для синхронных (точных) исключений, EPC содержит одно из следующего:

- виртуальный адрес команды, которая была прямой причиной исключения;
- виртуальный адрес команды перехода (branch или jump), непосредственно предшествующей исключению, если команда, вызвавшая исключение, находится в слоте задержки перехода и установлен бит bd в регистре cause.

Если установлен бит EXL в регистре Status, процессор не записывает адрес в регистр EPC при возникновении новых исключений. Однако, новое значение можно записать в EPC командой MTC0.

Формат регистра EPC

31	0
EPC	

Таблица 18.33. Описание полей регистра EPC

Номер бита	Поле	Тип	Описание	Сброс
31:0	EPC	R/W	Программный счетчик исключения	Не определено

18.2.7.17 Регистр PRId (Регистр 15 CP0, Select 0)

Регистр идентификации процессора (PRId) – это 32-х разрядный регистр, доступный только для чтения. Он содержит информацию, идентифицирующую изготовителя, опции изготовителя, идентификацию процессора, и версию процессора.

Формат регистра PRId

31 24	23 16	15 8	7 0
R	Company ID	Processor ID	Revision

Таблица 18.34. Описание полей регистра PRId

Номер бита	Поле	Тип	Описание	Сброс
	R	R	При чтении возвращается нуль	0
23:16	Company ID	R	Идентификация компании, которая проектировала или изготовляла процессор.	00001010
15:8	Processor ID	R	Идентификация типа процессора.	00100000
7:0	Revision	R	Номер версии процессора. Позволяет программам различать разные версии одного типа процессора.	00000001

18.2.7.18 Регистр Config (Регистр 16 CP0, Select 0)

Регистр Config определяет различную конфигурационную информацию, а также информацию о возможностях процессора. Большинство полей регистра Config инициализируется аппаратно при выполнении исключения Reset или имеет постоянное значение, и только поле K0 должно быть проинициализировано программно обработчиком исключения Reset.

Формат регистра Config

31	30 28	27 25	24 21	20 19	18 17	16	15	14 13	12 10	9 7	6 3	2 0
M	K23	KU	0	MDU	R	MM	BM	BE	AT	AR	MT	0	K0

Таблица 18.35. Описание полей регистра Config

Номер бита	Поле	Тип	Описание	Сброс
31	M	R	Этот бит аппаратно устанавливается в высокий уровень, указывая на наличие регистра Config1	1
30:28	K23	FM:R/W	Это поле управляет кэшируемостью адресных сегментов kseg2 и kseg3 в режиме FM. В режиме TLB не используется. См. табл.5.21.	FM:010
		TLB:R		TLB:000
27:25	KU	FM:R/W	Это поле управляет кэшируемостью адресных сегментов kuseg и useg в режиме FM. В режиме TLB не используется. См. табл.5.21.	FM:010
		TLB:R		TLB:000
24:21	0	0	Не используются	0
20	MDU	R	Тип MDU: итеративный умножитель и делитель	0
19	R	0	При чтении возвращается нуль	0
18:17	MM	R	Режим No Merging для 32 bit collapsing write buffer	11
16	BM	R	Тип передачи Burst: последовательный	0
15	BE	R	Режим endian: Little endian	0
14:13	AT	R	Тип архитектуры, реализованной процессором: MIPS32.	0
12:10	AR	R	Номер версии: 1	0
9:7	MT	R	Тип MMU: 1: Стандартный TLB (FM = 0) 3: Фиксированное отображение (FM = 1) 0, 2, 4-7: зарезервированы	TLB: 01 FM: 11
6:3	R	0	При чтении возвращается нуль	0
2:0	K0	R/W	Алгоритм когерентности для kseg0, см. Таблица 18.21.	010

Таблица 18.36. Атрибуты когерентности кэш

Значение C[5:3]	
0, 1, 3*, 4, 5, 6	Кэшируемая, некогерентная область
2*, 7	Некэшируемая область
* - Архитектура MIPS32 предусматривает только эти два значения. Остальные значения не используются и отображаются в используемые значения. Например, 0, 1, 4, 5 и 6 отображается в 3, а 7 – в 2.	

18.2.7.19 Регистр Config1 (Регистр 16 CP0, Select 1)

Регистр Config1 является дополнением к регистру Config и кодирует дополнительную информацию о возможностях процессора. Все поля регистра Config1 доступны только для чтения.

Формат регистра Config1

31	30 25	24 22	21 19	18 16	15 13	12 10	9 7	6 5	4	3	2	1	0
R	MMUSize	IS	IL	IA	DS	DL	DA	R	PC	WR	CA	EP	FP

Таблица 18.37. Описание полей Config1 регистра

Номер бита	Поле	Тип	Описание	Сброс
31	R	0	При чтении возвращается нуль	0
30:25	Размер MMU	R	Это поле содержит количество строк TLB минус 1. В режиме TLB возвращается 15 в десятичном формате, в режиме Fixed Mapping – 0	001111 (FM =0) 000000 (FM =1)
24:22	IS	R	Количество наборов кэш команд: резервная опция	111
21:19	IL	R	Размер строки кэш команд: 16 байт	011
18:16	IA	R	Тип кэш команд: Direct mapped	0
15:13	DS	R	Нет кэш данных	111
12:10	DL	R	Нет кэш данных	0
9:7	DA	R	Нет кэш данных	111
6:5	R	0	При чтении возвращается нуль	0
4	PC	R	Нет регистра Performance Counter	0
3	WR	R	Нет регистра WATCH	1
2	CA	R	Не реализовано	0
1	EP	R	EJTAG не реализован	1
0	FP	R	Нет плавающей арифметики	0

18.2.7.20 Регистр LLAddr – Load Linked Address (Регистр 17 CP0, Select 0)

Регистр LLAddr содержит физический адрес последней команды Load Linked (LL). Этот регистр используется только для диагностических целей.

Формат LLAddr регистра

31 28	27 0
0	Paddr[31:4]

Таблица 18.38. Описание полей LLAddr регистра

Номер бита	Поле	Тип	Описание	Сброс
31:28	0	0	При чтении возвращается нуль	0
27:0	Paddr[31:4]	R	Физический адрес последней команды LL	Не определено

18.2.7.21 Регистр PRnumber – Processor number register (Регистр 20 CP0, Select 0)

Регистр содержит номер данного процессора в составе кластера.

Таблица 18.39. Описание полей PRnumber регистра

Номер бита	Поле	Тип	Описание	Сброс
31:4	0	0	При чтении возвращается нуль	0
3:0	Num	R	Номер процессора в кластере	Не определено

18.2.7.22 Регистр ErrorEPC (Регистр 30 CP0, Select 0)

Доступный для чтения и записи, регистр ErrorEPC полностью подобен регистру EPC, но используется при возникновении исключений ошибок. Все биты регистра ErrorEPC значимы и должны перезаписываться. Регистр ErrorEPC также используется для сохранения значения счетчика команд при возникновении исключений Reset и немаскируемого прерывании (NMI).

Регистр ErrorEPC содержит виртуальный адрес, начиная с которого может возобновиться исполнение программы после обработки ошибочной ситуации.

Этот адрес может быть:

- виртуальным адресом команды, вызвавшей исключение;
- виртуальным адресом команды перехода (branch или jump), непосредственно предшествующей исключению, если команда, вызвавшая ошибку, находится в слоте задержки перехода.

В отличие от регистра EPC, для регистра ErrorEPC не имеется соответствующего признака слота задержки перехода.

Формат регистра ErrorEPC

31	0
ErrorEPC	

Таблица 18.40. Описание полей регистра ErrorEPC

Номер бита	Поле	Тип	Описание	Сброс
31:0	ErrorEPC	R/W	Счетчик команд при исключении ошибки	Не определен

Регистры WatchLo, WatchHi, Debug, DEPC, TagLo, DataLo, DeSave не реализованы

18.2.8 Кэш

18.2.8.1 Введение

CPU имеет кэш команд и кэш данных типа direct mapped объемом по 32 Кбайта. Кэш данных работает по протоколу write-through.

Кэш состоит из двух массивов – массива тэгов и массива данных. Кэш индексируется виртуально, поскольку для выбора соответствующей строки в обоих массивах используется виртуальный адрес. Это позволяет осуществлять доступ к кэш параллельно с преобразованием виртуального адреса в физический. Контроль осуществляется по физическому тэгу, так-так массив тэгов содержит физический, а не виртуальный адрес.

Пополнения кэш выполняются посредством четырехсловного буфера, в который поступают данные, полученные из памяти во время передачи 4-х ступенчатой пачкой (Burst). Критическое пропущенное слово всегда возвращается первым. До получения критического слова кэш блокируется, но во время активности на шине остальных 3-х ступеней Burst конвейер может продвигаться дальше.

18.2.8.2 Протокол кэш

18.2.8.3 Организация кэш

Кэш команд состоит из двух массивов – массива тэгов и массива данных. Кэш индексируется виртуально, поскольку для выбора соответствующей строки в обоих

массивах используется виртуальный адрес. Контроль осуществляется по физическому тэгу, так-так массив тэгов содержит физический, а не виртуальный адрес.

На Рисунок 18.21 представлен формат каждой строки массивов тэгов и данных. Тэговая строка содержит 22 старших бита физического адреса (биты [31:10]) и бит валидности.

Строка данных содержит 4 32-х разрядных слова – всего 16 байт.

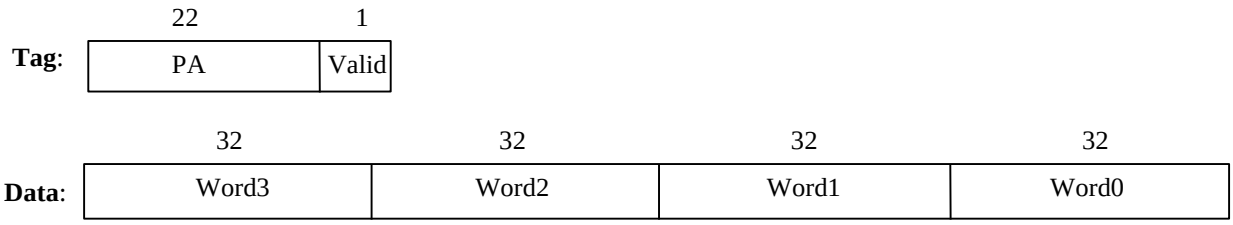


Рисунок 18.21. Формат массива кэш

18.2.8.4 Атрибуты кэшируемости.

В данной версии реализовано только два атрибута. Область может быть либо кэшируемой, либо некэшируемой (см. Таблица 18.36)

18.2.9 Карта памяти VCPU

Вид физической памяти со стороны VCPU аналогичен виду карты физической памяти для всей микросхемы (приведена в Таблице 3.1). Однако, есть ряд областей доступных для VCPU, отображение которых замещает области в общей карте памяти микросхемы. Эти области указаны в Таблица 18.41, это область ROM памяти для процессора, где хранится код начальной загрузки и область памяти быстрого доступа CRAM.

Таблица 18.41. Специальные области памяти для VCPU

Диапазон адресов	Название области	Объем области, Кбайт	Доступ
0x1000_0000 – 0x1000_7FFF	Память CRAM	32	Доступ для VCPU по внутренней шине процессора, для остальных область доступна через другие адреса – зеркала (см. Таблица 3.1 и Таблица 5.10)
0x1FC0_0000 – 0x1FC0_7FFF	Внешняя память (ПЗУ)	32	Эксклюзивный доступ только для VCPU
0x1FC0_0000 – 0x1FFF_FFFF	Резерв	64	-

Общий объем адресуемой физической памяти VCPU ограничен 4 Гбайтами, однако он может быть разбит на ряд регионов, которые можно переадресовывать на адреса старше 4 Гбайт в физической памяти микросхемы с помощью вторичного преобразования адреса в блоке VCU VMMU (см. Раздел 18.1).

18.2.10 Блок отладки VCPU OnCD

18.2.10.1 Назначение

Встроенные средства отладки – OnCD (On Chip Debugger for Multicore) предназначены для построения эффективных систем отладки программ. OnCD представляет собой набор отладочных регистров, через которые осуществляется функционал отладки. Регистры могут быть доступны через выделенный интерфейс JTAG (см. Раздел 3.2.16), либо через обращение к области памяти VCU VCPU OnCD (см Таблицу 5.10).

18.2.10.2 Регистры OnCD

Перечень регистров OnCD и VCPU приведен в Таблица 18.42.

Таблица 18.42. Перечень регистров OnCD в VCPU

Условное обозначение регистра	Название регистра	Адрес (offset)	Вид доступа
IR	Регистр команд TAP контроллера OnCD	0x0	r/w
IDcode	Регистр ID code OnCD	0x4	r

Условное обозначение регистра	Название регистра	Адрес (offset)	Вид доступа
OSCR	Регистр состояния и управления OnCD	0x8	r/w
OMBC	Регистр счетчика контрольных точек OnCD	0xc	r/w
OMLR0	Регистр первой границы адреса сравнения OnCD	0x10	r/w
OMLR1	Регистр второй границы адреса сравнения OnCD	0x14	r/w
OBCR	Регистр управления остановом OnCD	0x18	r/w
IRdec	Регистр команд RISC	0x1c, 0x21c	r/w
OTC	Регистр счетчика трассы OnCD	0x20	r/w
PCdec	Регистр адреса команды RISC на стадии декодирования	0x24	r/w
PCexe	Регистр адреса команды RISC на стадии выполнения	0x28	r/w
PCmem	Регистр адреса команды RISC на стадии обмена с памятью	0x2c	r/w
PC	Регистр адреса команды RISC на стадии выборки	0x30, 0x230	r/w
OMAR	Регистр адреса памяти OnCD	0x34	r/w
OMDR	Регистр данных памяти OnCD	0x38	r/w
MEM	Регистр запуска контроллера памяти OnCD	0x3c	w
PCwb	Регистр адреса команды RISC на стадии записи в RF	0x40	r/w
RegF	Регистр связи с напрямую неадресуемыми объектами RISC	0x44, 0x244	r/w
GO	Регистр пуска RISC процессора на выполнение программы	0x48	w

18.2.10.3 Регистр IR

Регистр IR предназначен для перевода процессора в режим отладки, путем записи в него значения 0x4. При чтении этого регистра возвращается значение состояния процессора, если считалось значение 0x5, то процессор находится в режиме отладки, если считалось значение 0x1, то не находится.

18.2.10.4 Регистр IDcode

Регистр IDcode доступен только по чтению. Его значение 0x601197CF.

18.2.10.5 Регистр IRdec

Для получения данных при чтении IRdec, необходимо предварительно сделать запись по адресу 0x1c или 0x21c. После записи значения по адресу 0x1c содержимое IRdec из RISC будет доступно при чтении по адресу 0x1c, при этом новое значение будет перезаписано в IRdec в RISC. После записи по адресу 0x21c содержимое IRdec из RISC так же будет доступно при чтении по адресу 0x1c, но новое значение не будет записано в IRdec в RISC.

18.2.10.6 Регистр PC

Для получения данных при чтении PC, необходимо предварительно сделать запись по адресу 0x30 или 0x230. После записи значения по адресу 0x30 содержимое PC из RISC будет доступно при чтении по адресу 0x30, при этом новое значение будет перезаписано в PC в RISC. После записи по адресу 0x230 содержимое PC из RISC так же будет доступно при чтении по адресу 0x30, но новое значение не будет записано в PC в RISC.

18.2.10.7 Регистр RegF

Для получения данных при чтении RegF, необходимо предварительно сделать запись по адресу 0x44 или 0x244. После записи значения по адресу 0x44 содержимое RegF из RISC будет доступно при чтении по адресу 0x44, при этом новое значение будет перезаписано в RegF в RISC. После записи по адресу 0x244 содержимое RegF из RISC так же будет доступно при чтении по адресу 0x44, но новое значение не будет записано в RegF в RISC.

18.2.10.8 Регистры PCdec, PCexe, PCmem, PCwb

Для получения данных при чтении этих регистров, необходимо предварительно сделать запись любого значения по соответствующему адресу нужного регистра. Значения этих регистров при записи не перезаписываются в RISC.

18.2.10.9 Регистр MEM

Регистр MEM доступен только по записи. Для запуска контроллера памяти модуля OnCD, в этот регистр необходимо записать значение 0xd.

18.2.10.10 Регистр GO

Регистр GO доступен только по записи. Для запуска процессора на выполнение программы, в этот регистр необходимо записать значение {nStep, push_IRdec, mod_Step, 1`b0, 4`hf}.

Остальные регистры модуля OnCD пишутся и читаются напрямую. Более подробное описание модуля OnCD и работы с ним находится в документе «Встроенные средства отладки программ сигнальных микроконтроллеров мультикор_F (OnCD_F)».

19. СЛУЖЕБНЫЙ КОНТРОЛЛЕР ПРЕРЫВАНИЙ (QLIC)

Для распределения прерываний в сервисной подсистеме и подсистеме SDR используется контроллер QLIC. Контроллер соответствует спецификации RISC-V Privileged Architecture версии 1.9.1 и в данной реализации поддерживает 128 источников внешних прерываний и 18 объектов назначения прерываний. Принципиальная схема организации контроллера QLIC показаны на Рисунок 19.1.

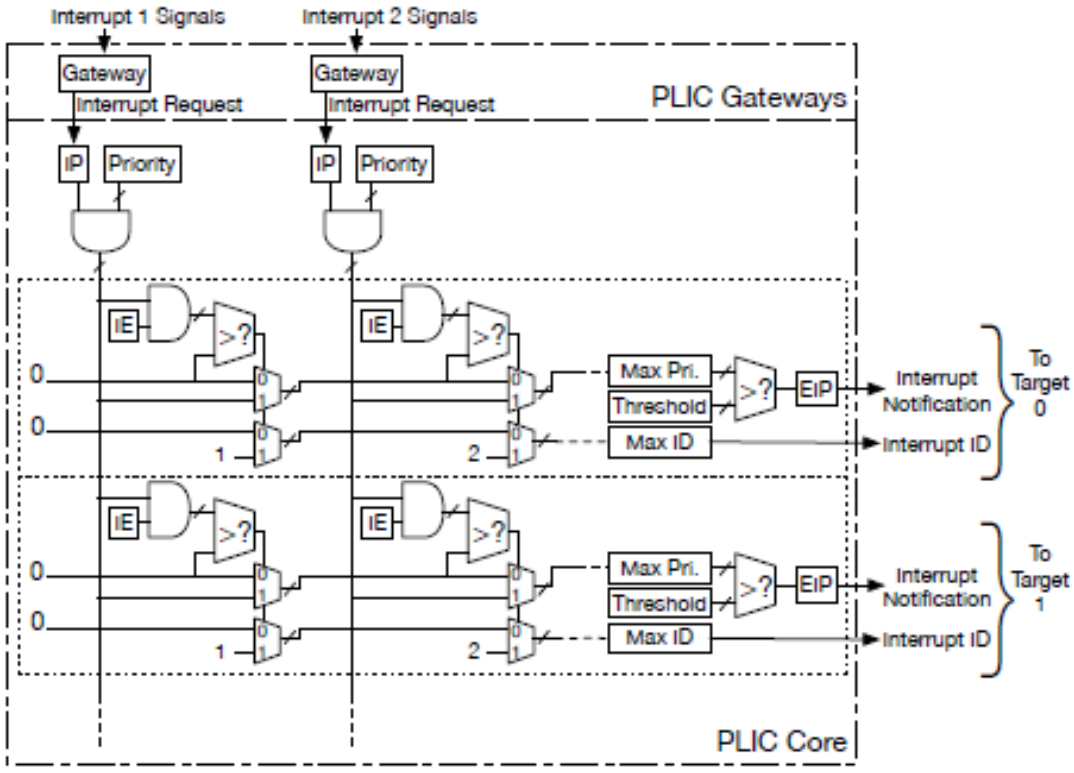


Рисунок 19.1. Принципиальная схема QLIC

Карта памяти контроллера QLIC показана в Таблица 19.1. Распределение входных прерываний описано в Разделе 3.5. Распределение выходных прерываний, а также базовые адреса контроллеров QLIC описаны в главах соответствующих им подсистем «Служебная подсистема» и «Подсистема SDR».

Таблица 19.1. Сводная таблица регистров QLIC

Смещение	Обозначение	Описание
Регистры управления системой синхронизации		
0x0_0000	-	Резерв
0x0_0004	S1_PRI	Приоритет 1-го прерывания
0x0_0008	S2_PRI	Приоритет 2-го прерывания
0x0_000c	S3_PRI	Приоритет 3-го прерывания
0x0_0010	S4_PRI	Приоритет 4-го прерывания
0x0_0014	S5_PRI	Приоритет 5-го прерывания
0x0_0018	S6_PRI	Приоритет 6-го прерывания
0x0_001c	S7_PRI	Приоритет 7-го прерывания
0x0_0020	S8_PRI	Приоритет 8-го прерывания

Смещение	Обозначение	Описание
0x0_0024	S9_PRI	Приоритет 9-го прерывания
0x0_0028	S10_PRI	Приоритет 10-го прерывания
0x0_002c	S11_PRI	Приоритет 11-го прерывания
0x0_0030	S12_PRI	Приоритет 12-го прерывания
0x0_0034	S13_PRI	Приоритет 13-го прерывания
0x0_0038	S14_PRI	Приоритет 14-го прерывания
0x0_003c	S15_PRI	Приоритет 15-го прерывания
0x0_0040	S16_PRI	Приоритет 16-го прерывания
0x0_0044	S17_PRI	Приоритет 17-го прерывания
0x0_0048	S18_PRI	Приоритет 18-го прерывания
0x0_004c	S19_PRI	Приоритет 19-го прерывания
0x0_0050	S20_PRI	Приоритет 20-го прерывания
0x0_0054	S21_PRI	Приоритет 21-го прерывания
0x0_0058	S22_PRI	Приоритет 22-го прерывания
0x0_005c	S23_PRI	Приоритет 23-го прерывания
0x0_0060	S24_PRI	Приоритет 24-го прерывания
0x0_0064	S25_PRI	Приоритет 25-го прерывания
0x0_0068	S26_PRI	Приоритет 26-го прерывания
0x0_006c	S27_PRI	Приоритет 27-го прерывания
0x0_0070	S28_PRI	Приоритет 28-го прерывания
0x0_0074	S29_PRI	Приоритет 29-го прерывания
0x0_0078	S30_PRI	Приоритет 30-го прерывания
0x0_007c	S31_PRI	Приоритет 31-го прерывания
0x0_0080	S32_PRI	Приоритет 32-го прерывания
0x0_0084	S33_PRI	Приоритет 33-го прерывания
0x0_0088	S34_PRI	Приоритет 34-го прерывания
0x0_008c	S35_PRI	Приоритет 35-го прерывания
0x0_0090	S36_PRI	Приоритет 36-го прерывания
0x0_0094	S37_PRI	Приоритет 37-го прерывания
0x0_0098	S38_PRI	Приоритет 38-го прерывания
0x0_009c	S39_PRI	Приоритет 39-го прерывания
0x0_00a0	S40_PRI	Приоритет 40-го прерывания
0x0_00a4	S41_PRI	Приоритет 41-го прерывания
0x0_00a8	S42_PRI	Приоритет 42-го прерывания
0x0_00ac	S43_PRI	Приоритет 43-го прерывания
0x0_00b0	S44_PRI	Приоритет 44-го прерывания
0x0_00b4	S45_PRI	Приоритет 45-го прерывания
0x0_00b8	S46_PRI	Приоритет 46-го прерывания
0x0_00bc	S47_PRI	Приоритет 47-го прерывания
0x0_00c0	S48_PRI	Приоритет 48-го прерывания
0x0_00c4	S49_PRI	Приоритет 49-го прерывания
0x0_00c8	S50_PRI	Приоритет 50-го прерывания
0x0_00cc	S51_PRI	Приоритет 51-го прерывания
0x0_00d0	S52_PRI	Приоритет 52-го прерывания
0x0_00d4	S53_PRI	Приоритет 53-го прерывания
0x0_00d8	S54_PRI	Приоритет 54-го прерывания
0x0_00dc	S55_PRI	Приоритет 55-го прерывания
0x0_00e0	S56_PRI	Приоритет 56-го прерывания
0x0_00e4	S57_PRI	Приоритет 57-го прерывания
0x0_00e8	S58_PRI	Приоритет 58-го прерывания
0x0_00ec	S59_PRI	Приоритет 59-го прерывания
0x0_00f0	S60_PRI	Приоритет 60-го прерывания
0x0_00f4	S61_PRI	Приоритет 61-го прерывания
0x0_00f8	S62_PRI	Приоритет 62-го прерывания
0x0_00fc	S63_PRI	Приоритет 63-го прерывания
0x0_0100	S64_PRI	Приоритет 64-го прерывания
0x0_0104	S65_PRI	Приоритет 65-го прерывания

Смещение	Обозначение	Описание
0x0_0108	S66_PRI	Приоритет 66-го прерывания
0x0_010c	S67_PRI	Приоритет 67-го прерывания
0x0_0110	S68_PRI	Приоритет 68-го прерывания
0x0_0114	S69_PRI	Приоритет 69-го прерывания
0x0_0118	S70_PRI	Приоритет 70-го прерывания
0x0_011c	S71_PRI	Приоритет 71-го прерывания
0x0_0120	S72_PRI	Приоритет 72-го прерывания
0x0_0124	S73_PRI	Приоритет 73-го прерывания
0x0_0128	S74_PRI	Приоритет 74-го прерывания
0x0_012c	S75_PRI	Приоритет 75-го прерывания
0x0_0130	S76_PRI	Приоритет 76-го прерывания
0x0_0134	S77_PRI	Приоритет 77-го прерывания
0x0_0138	S78_PRI	Приоритет 78-го прерывания
0x0_013c	S79_PRI	Приоритет 79-го прерывания
0x0_0140	S80_PRI	Приоритет 80-го прерывания
0x0_0144	S81_PRI	Приоритет 81-го прерывания
0x0_0148	S82_PRI	Приоритет 82-го прерывания
0x0_014c	S83_PRI	Приоритет 83-го прерывания
0x0_0150	S84_PRI	Приоритет 84-го прерывания
0x0_0154	S85_PRI	Приоритет 85-го прерывания
0x0_0158	S86_PRI	Приоритет 86-го прерывания
0x0_015c	S87_PRI	Приоритет 87-го прерывания
0x0_0160	S88_PRI	Приоритет 88-го прерывания
0x0_0164	S89_PRI	Приоритет 89-го прерывания
0x0_0168	S90_PRI	Приоритет 90-го прерывания
0x0_016c	S91_PRI	Приоритет 91-го прерывания
0x0_0170	S92_PRI	Приоритет 92-го прерывания
0x0_0174	S93_PRI	Приоритет 93-го прерывания
0x0_0178	S94_PRI	Приоритет 94-го прерывания
0x0_017c	S95_PRI	Приоритет 95-го прерывания
0x0_0180	S96_PRI	Приоритет 96-го прерывания
0x0_0184	S97_PRI	Приоритет 97-го прерывания
0x0_0188	S98_PRI	Приоритет 98-го прерывания
0x0_018c	S99_PRI	Приоритет 99-го прерывания
0x0_0190	S100_PRI	Приоритет 100-го прерывания
0x0_0194	S101_PRI	Приоритет 101-го прерывания
0x0_0198	S102_PRI	Приоритет 102-го прерывания
0x0_019c	S103_PRI	Приоритет 103-го прерывания
0x0_01a0	S104_PRI	Приоритет 104-го прерывания
0x0_01a4	S105_PRI	Приоритет 105-го прерывания
0x0_01a8	S106_PRI	Приоритет 106-го прерывания
0x0_01ac	S107_PRI	Приоритет 107-го прерывания
0x0_01b0	S108_PRI	Приоритет 108-го прерывания
0x0_01b4	S109_PRI	Приоритет 109-го прерывания
0x0_01b8	S110_PRI	Приоритет 110-го прерывания
0x0_01bc	S111_PRI	Приоритет 111-го прерывания
0x0_01c0	S112_PRI	Приоритет 112-го прерывания
0x0_01c4	S113_PRI	Приоритет 113-го прерывания
0x0_01c8	S114_PRI	Приоритет 114-го прерывания
0x0_01cc	S115_PRI	Приоритет 115-го прерывания
0x0_01d0	S116_PRI	Приоритет 116-го прерывания
0x0_01d4	S117_PRI	Приоритет 117-го прерывания
0x0_01d8	S118_PRI	Приоритет 118-го прерывания
0x0_01dc	S119_PRI	Приоритет 119-го прерывания
0x0_01e0	S120_PRI	Приоритет 120-го прерывания
0x0_01e4	S121_PRI	Приоритет 121-го прерывания
0x0_01e8	S122_PRI	Приоритет 122-го прерывания

Смещение	Обозначение	Описание
0x0_01ec	S123_PRI	Приоритет 123-го прерывания
0x0_01f0	S124_PRI	Приоритет 125-го прерывания
0x0_01f4	S125_PRI	Приоритет 125-го прерывания
0x0_01f8	S126_PRI	Приоритет 126-го прерывания
0x0_01fc	S127_PRI	Приоритет 127-го прерывания
0x0_0200-0x0_2ffc	-	Резерв
0x0_0800	PEND_0_31	Индикация запросов на обработку от источников 0 – 31
0x0_0804	PEND_32_63	Индикация запросов на обработку от источников 32 – 63
0x0_0808	PEND_64_95	Индикация запросов на обработку от источников 64 – 95
0x0_080c	PEND_96_127	Индикация запросов на обработку от источников 96 – 127
0x0_0810 – 0x0_1ffff	-	Резерв
0x0_1000	TARG0_ENS	Разрешение 0-го выходного прерывания
0x0_1080	TARG1_ENS	Разрешение 1-го выходного прерывания
0x0_1100	TARG2_ENS	Разрешение 2-го выходного прерывания
0x0_1180	TARG3_ENS	Разрешение 3-го выходного прерывания
0x0_1200	TARG4_ENS	Разрешение 4-го выходного прерывания
0x0_1280	TARG5_ENS	Разрешение 5-го выходного прерывания
0x0_1300	TARG6_ENS	Разрешение 6-го выходного прерывания
0x0_1380	TARG7_ENS	Разрешение 7-го выходного прерывания
0x0_1400	TARG8_ENS	Разрешение 8-го выходного прерывания
0x0_1480	TARG9_ENS	Разрешение 9-го выходного прерывания
0x0_1500	TARG10_ENS	Разрешение 10-го выходного прерывания
0x0_1580	TARG11_ENS	Разрешение 11-го выходного прерывания
0x0_1600	TARG12_ENS	Разрешение 12-го выходного прерывания
0x0_1680	TARG13_ENS	Разрешение 13-го выходного прерывания
0x0_1700	TARG14_ENS	Разрешение 14-го выходного прерывания
0x0_1780	TARG15_ENS	Разрешение 15-го выходного прерывания
0x0_1800	TARG16_ENS	Разрешение 16-го выходного прерывания
0x0_1880	TARG17_ENS	Разрешение 17-го выходного прерывания
0x0_1900 - 0x0_FFFC	-	Резерв
0x1_0000	TARG0_THD	Регистр порогового приоритета для 0-го выходного прерывания
0x1_0004	TARG0_CC	Регистр заявки/завершения 0-го выходного прерывания
0x1_1000	TARG1_THD	Регистр порогового приоритета для 1-го выходного прерывания
0x1_1004	TARG1_CC	Регистр заявки/завершения 1-го выходного прерывания
0x1_2000	TARG2_THD	Регистр порогового приоритета для 2-го выходного прерывания
0x1_2004	TARG2_CC	Регистр заявки/завершения 2-го выходного прерывания
0x1_3000	TARG3_THD	Регистр порогового приоритета для 3-го выходного прерывания
0x1_3004	TARG3_CC	Регистр заявки/завершения 3-го выходного прерывания
0x1_4000	TARG4_THD	Регистр порогового приоритета для 4-го выходного прерывания
0x1_4004	TARG4_CC	Регистр заявки/завершения 4-го выходного прерывания
0x1_5000	TARG5_THD	Регистр порогового приоритета для 5-го выходного прерывания
0x1_5004	TARG5_CC	Регистр заявки/завершения 5-го выходного прерывания
0x1_6000	TARG6_THD	Регистр порогового приоритета для 6-го выходного прерывания
0x1_6004	TARG6_CC	Регистр заявки/завершения 6-го выходного прерывания
0x1_7000	TARG7_THD	Регистр порогового приоритета для 7-го выходного прерывания
0x1_7004	TARG7_CC	Регистр заявки/завершения 7-го выходного прерывания
0x1_8000	TARG8_THD	Регистр порогового приоритета для 8-го выходного прерывания
0x1_8004	TARG8_CC	Регистр заявки/завершения 8-го выходного прерывания
0x1_9000	TARG9_THD	Регистр порогового приоритета для 9-го выходного прерывания
0x1_9004	TARG9_CC	Регистр заявки/завершения 9-го выходного прерывания
0x1_a000	TARG10_THD	Регистр порогового приоритета для 10-го выходного прерывания

Смещение	Обозначение	Описание
0x1_a004	TARG10_CC	Регистр заявки/завершения 10-го выходного прерывания
0x1_b000	TARG11_THD	Регистр порогового приоритета для 11-го выходного прерывания
0x1_b004	TARG11_CC	Регистр заявки/завершения 11-го выходного прерывания
0x1_c000	TARG12_THD	Регистр порогового приоритета для 12-го выходного прерывания
0x1_c004	TARG12_CC	Регистр заявки/завершения 12-го выходного прерывания
0x1_d000	TARG13_THD	Регистр порогового приоритета для 13-го выходного прерывания
0x1_d004	TARG13_CC	Регистр заявки/завершения 13-го выходного прерывания
0x1_e000	TARG14_THD	Регистр порогового приоритета для 14-го выходного прерывания
0x1_e004	TARG14_CC	Регистр заявки/завершения 14-го выходного прерывания
0x1_f000	TARG15_THD	Регистр порогового приоритета для 15-го выходного прерывания
0x1_f004	TARG15_CC	Регистр заявки/завершения 15-го выходного прерывания
0x2_0000	TARG0_THD	Регистр порогового приоритета для 16-го выходного прерывания
0x2_0004	TARG0_CC	Регистр заявки/завершения 16-го выходного прерывания
0x2_1000	TARG0_THD	Регистр порогового приоритета для 17-го выходного прерывания
0x2_1004	TARG0_CC	Регистр заявки/завершения 17-го выходного прерывания

19.1 Источники прерываний

Контроллер QLIC распределяет 128 источников прерываний. Все источники прерываний являются level sensitive, т.е. работают по уровню, при этом активным уровнем является 1. ID источника определяется как n в обозначении Sn_PRI соответствующего регистра приоритета прерывания в Таблица 19.1.

19.2 Приоритеты прерываний и регистры Sn_PRI

Каждому источнику прерываний может быть назначен приоритет от 0 до 7, с помощью записи в соответствующий регистр Sn_PRI. Приоритет 0 означает фактическое маскирование данного прерывания. Источник прерывания с большим приоритетом имеет преимущество в рамках определенного выходного прерывания для объекта назначения. В случае если заданы равные приоритеты, преимущество имеет прерывание с меньшим ID источника.

Таблица 19.2. Формат регистра Sn_PRI

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:3	-	Резерв	R	н.д.
2:0	PRI	Приоритет прерывания	RW	н.д.

19.3 Регистрация запросов на обработку прерываний и регистры PEND_n_m

Текущий статус запроса на обработку прерываний от источников может быть прочитан из массива 32 x 32 регистров PEND_n_m. Каждый из этих 32-разрядных регистров содержит бит с номером равным ID mod 32 индицирующий регистрацию запроса на прерывания в диапазоне ID источников от n до m. Бит 0 регистра PEND_0_32, представляющий не существующий источник 0, всегда читается нулем.

Таблица 19.3. Формат регистра PEND_n_m

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31	PEND_m	Зарегистрирован запрос на прерывание от источника m mod 32	R	н.д.
k	PEND_k	Зарегистрирован запрос на прерывание от источника k mod 32	R	н.д.
0	PEND_n	Зарегистрирован запрос на прерывание от источника n mod 32	R	н.д.

19.4 Разрешение прерываний для объекта назначения и регистры TARGn_ENS

Для каждого выходного прерывания объекта назначения, каждое входное прерывание от источника может быть разрешено установкой соответствующего бита в регистрах TARGn_ENS. Биты разрешения для каждого объекта назначения n сгруппированы в массивы 32 x 32 слов, соответвтвующие регистрам TARGn_ENS. Каждый массив (регистр TARGn_ENS) организован таким же образом, как и биты индикации на обработку прерываний. Для каждого объекта назначения бит 0 слова 0 представляет не существующий источник с ID 0 и привязан к нулю.

Таблица 19.4. Формат регистра m в массиве TARGn_ENS

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31	EN_32*m	Разрешение на прерывание от источника 32*m mod 32 для объекта назначения n	RW	н.д.
k	EN_k*m	Разрешение на прерывание от источника k*m mod 32 для объекта назначения n	RW	н.д.
0	EN_m	Разрешение на прерывание от источника m mod 32 для объекта назначения n	RW	н.д.

19.5 Пороговый приоритет для прерывания и регистры TARGn_THD

Существует возможность задать для каждого объекта назначения пороговый приоритет, прерывания от источников с приоритетом заданным ниже порогового не могут направляться на этот объект назначения.

Таблица 19.5. Формат регистра TARGn_THD

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:3	-	Резерв	R	н.д.
2:0	THD	Пороговый приоритет для объекта назначения n	RW	н.д.

19.6 Заявка на обработку прерывания и регистры TARGn_CC

Каждый объект назначения прерывания n может выставить заявку на обработку этого прерывания прочитав свой регистр заявки/завершения прерывания TARGn_CC, прочитанное значение будет равно ID прерывания с наивысшим приоритетом, запрос от которого зарегистрирован, либо нулю если зарегистрированного запроса нет. Успешная заявка также атомарно снимает соответствующий бит в регистрах регистрации запросов на прерывания PEND_n_m.

Таблица 19.6. Формат регистра TARGn_CC

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:0	CC	По чтению выполняется заявка на обработку прерывания, в случае успешной заявки читается ID зарегистрированного прерывания с наивысшим приоритетом, в случае если зарегистрированных на обработку прерываний нет читается нулем. Запись значения N сигнализирует логике QLIC о завершении обработки прерывания с ID равным N.	RW	н.д.

19.7 Завершение обработки прерывания

Объект назначения прерывания сигнализирует о том, что он завершил его обработку записью значения ID, равного тому что было получено при заявке, в регистр TARGn_CC. Факт такой записи позволяет логике контроллера QLIC регистрировать новые запросы прерываний от источника с записанным ID. Проверка на соответствие прочитанного ранее и записанного ID не производиться. Если записанное значение не соответствует прерываниям, разрешенным в регистрах TARGn_ENS для данного объекта назначения, то запись игнорируется.

20. НЕЙРОСЕТЕВОЙ АКСЕЛЕРАТОР С ОПТИМИЗАЦИЕЙ ДЛЯ СИГНАТУРНОГО АНАЛИЗА И ОБРАБОТКИ СИГНАЛОВ (DSP)

20.1 Общая информация

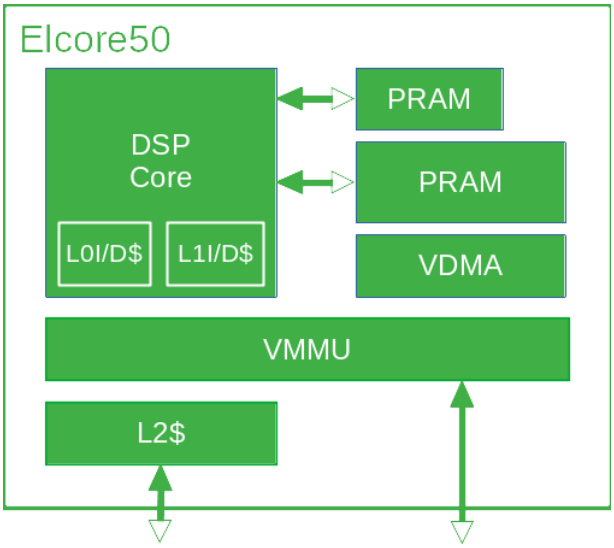


Рисунок 20.1. Ядро Elcore50

Нейросетевой акселератор с оптимизацией для сигнатурного анализа и обработки сигналов разработан на базе отечественного доверенного ядра Elcore 50.

Ядро Elcore50 является универсальным процессором с инновационной архитектурой, поддерживающей как универсальные возможности (CPU), так и возможности векторной обработки на основе SIMD расширения.

Ядро предназначено для широкого спектра применений, включая приложения, требующие программную обработку сетевых пакетов с использованием возможностей сигнатурного анализа для телекоммуникационных и иных приложений, а также сигнальную обработку (DSP) для применений микросхемы К1892ВМ21Я в создания базовых станций совместно с микросхемами «Гиперком» и «Гиперком МК».

Таким образом, ядро поддерживает области применения микросхемы «Гиперком У» в качестве не только малопотребляющего управляющего инфраструктурного процессора (Control Plane), но возможного выполнения функций Data Plane на его ресурсах.

Ядро Elcore50 является продолжением и расширением линейки DSP ядер «Elcore в которую входят ядра Elcore 12, Elcore24, Elcore 30, Elcore30M, Elcore40 и ряд их модификаций. Линейка Elcore предназначена для вычислительно-интенсивных задач цифровой обработки сигналов и мультимедийных приложений. Ядро Elcore50 совмещают

высокую производительность с низким энергопотреблением, эффективную поддержку компиляторов и высокую плотность кода во VLIW пакетах.

По сравнению с предыдущим поколением ядро Elcore 50 претерпело глубокую переработку с максимальным сохранением преемственности. Основные изменения следующие:

- Улучшена ортогональность упакованных во VLIW команд, а также количество команд в одном пакете
- Улучшена эффективность управляющего кода
- Улучшена поддержка компилятора
- Введена предикатная система с маскированием векторных операций
- Добавлен независимый векторный регистровый файл для специальных команд векторной и матричной обработки изображений
- Увеличены размеры кэшей L1DC и L1IC, и памятей XYRAM и PRAM
- Улучшен механизм отладки

Таблица 20.1. Основные характеристики ядра Elcore50

Параметр		Значение	
Базовая частота		не менее 340 МГц*	
Поддерживаемые форматы		Int8, Int16, Int32, Int64, FP16, FP32, FP64	
Пиковая производительность операций на такт	Скалярные	Арифметико-логические	Все форматы: 4
		Умножения	Все форматы: 2
	Векторные (над регистрами)	Арифметико-логические	Все форматы: 4
		Умножения с записью результатов в регистры	Int8: 128 Int16: 64 Int32: 32 Int64: 8 FP16: 64 FP32: 32 FP64: 16
	Над матрицами и фильтрация (над аккумуляторами)	Умножения матриц или сумма произведений с накоплением в аккумуляторах	Int8: 128 Int16: 64 Int32: 32 FP16: 1024 FP32: 256 FP64: 64
		Фильтрация с накоплением результата в аккумуляторах	Int8: 1024 Int16: 256 FP16: 1024 FP32: 256
Число Load/Store операций за такт		2x(8 байт скалярных или 64 байта векторных)	
Конфигурация L0I$		512 бит, 16-путевая ассоциативность	
Конфигурация L0D$		4 Кбит, прямого отображения	

Конфигурация L1I$	16 Кбит, 4-путевая ассоциативность
Конфигурация L1D$	16 Кбит, 4-путевая ассоциативность
Конфигурация L2$	До 512 Кбит, 16-путевая ассоциативность
Локальная память данных XYRAM	До 512 Кбит
Локальная память программ PRAM	64 Кбит
Пропускная способность XYRAM	128 байт/такт
Пропускная способность внешней памяти	16 байт/такт
Пропускная способность VDMA	16 байт/такт

* Подлежит уточнению после получения и измерения макетных образцов микросхемы К1892ВМ21Я.

20.2Описание

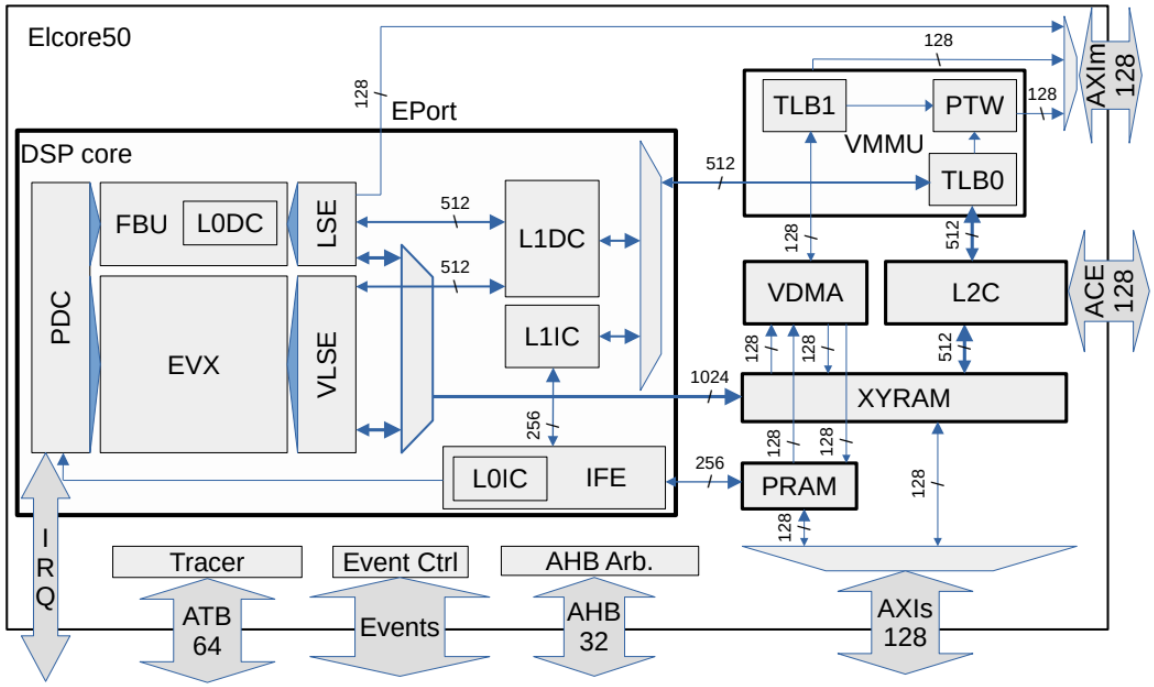


Рисунок 20.2. Структурная схема DSP ядра Elcore50

В состав DSP ядра Elcore50 входят:

- Elcore50 – верхний уровень DSP ядра, включающий некоторые вспомогательные блоки
- AHB Arb. – арбитр обслуживания обращений в регистры DSP ядра по шине AHB
- Event Ctrl. – блок поддержки аппаратных событий.
- Tracer – блок формирования аппаратной трассы DSP ядра.
- VDMA (Virtual Direct Memory Access) – блок пересылок между локальной памятью PRAM/XYRAM и внешними адресами без участия DSP ядра. Состоит из 4-ёх независимых каналов.
- VMMU (Virtual Memory Mapping Unit) – блок трансляции адресов из виртуальных в физические и проверки прав доступа. Состоит из:
  - PTW (Page Table Walker) – блок для подгрузки записей PTE, содержащих информацию для работы VMMU.
  - TLB0 (Table Lookup Block) – блок для обработки приходящих обращений и формирования запросов к PTW. Обслуживает запросы от DSP. Транслирует 32 битный виртуальный адрес в 41 битный физический.
  - TLB1 – блок для обработки приходящих обращений и формирования запросов к PTW. Обслуживает запросы от VDMA. Транслирует 41 битный виртуальный адрес в 41 битный физический.
- L2C – контроллер кэш памяти 2-ого уровня PIPT типа (Physical Indexed Physical Tagged, физические индекс и тэги). Содержит память тэгов, а в качестве памяти данных использует XYRAM.
- PRAM – локальная память программ. Размер:64Кбайт.
- XYRAM локальная память данных. Размер:512Кбайт. Состоит из двух групп банков, к которым можно обращаться в блочном и интерлив режимах.
- DSP core – основной уровень DSP ядра. Работает на одной частоте – CLK0.
  - L1DC – кэш данных память 1-ого уровня. Размер: 16Кбайт. Число путей: 4. Размер строки: 64 байта.
  - L1IC – кэш память инструкций 1-ого уровня. Размер: 16Кбайт. Число путей: 4. Размер строки: 64 байта.
  - L0DC – кэш память данных 0-ого уровня. Размер: 4 Кбайта. Число путей: 1. Размер строки: 32 байта.
  - L0IC – кэш память инструкций 0-ого уровня. Размер: 512 байт. Число путей: 16. Размер строки: 32 байта.
  - IFE – блок выборки VLIW инструкций из памяти. Так же выполняет предвыборку и поддерживает аппаратные DO циклы.

- PDC – блок декодеров VLIW инструкций.
- FBU – блок, выполняющий скалярные операции. Работает над 1-8 байтами данных. До 4-ёх инструкций за такт.
- EVX – блок, выполняющий векторные и матричные операции. SIMD из 8-ми секций с обработкой до 8-ми байт в каждой. До 4-ёх инструкций за такт.
- LSE (Load-Store Engine) – блок, выполняющий скалярные пересылки с памятью.
- VLSE (Vector Load-Store Engine) – блок, выполняющий векторные пересылки с памятью, в том числе векторно-индексные операции с локальной памятью XYRAM.

Ядро DSP подключается к системным коммутаторам по портам:

- AXIm (DMA) – внешний мастер порт AXI3 для некэшируемых обращений. Имеет ширину данных 128 бит и ширину адреса 41 бит. Работает на частоте AMCLK (BBD_ACLK). Поддерживает реордеринг уникальных ID и не поддерживает интерлив данных по чтению.
- AXIs (SRAM) – внешний слэйв порт AXI3 для обращений в локальную память PRAM и XYRAM. Имеет ширину данных 128 бит и ширину адреса 32 бита. Работает на частоте ASCLK (BBD_ACLK). Не делает реордеринг и интерлив, не поддерживает интерлив по записи.
- ACE (L2) – внешний мастер порт (ACE Lite + AXI4) для кэшируемых обращений. Имеет ширину данных 128 бит и ширину адреса 41 бит. Работает на частоте ядра CLK0 (DSP0_CLK и DSP1_CLK). Поддерживает IO coherency, реордеринг уникальных ID и не поддерживает интерлив данных по чтению.
- AHB (Regs) – внешний слэйв порт для обращений в регистры ядра Elcore50. Имеет ширину данных 32 бита и ширину адреса 32 бита. Работает на частоте HCLK (CFG_CLK).
- ATB – внешний мастер порт для выгрузки аппаратной трассы. Имеет ширину данных 32 бита. Работает на частоте ATCLK.
- EVENTS – системный порт аппаратных событий.
- IRQ – системный порт прерываний.
- EPort – внутренний порт некэшируемых обращений от DSP ядра.

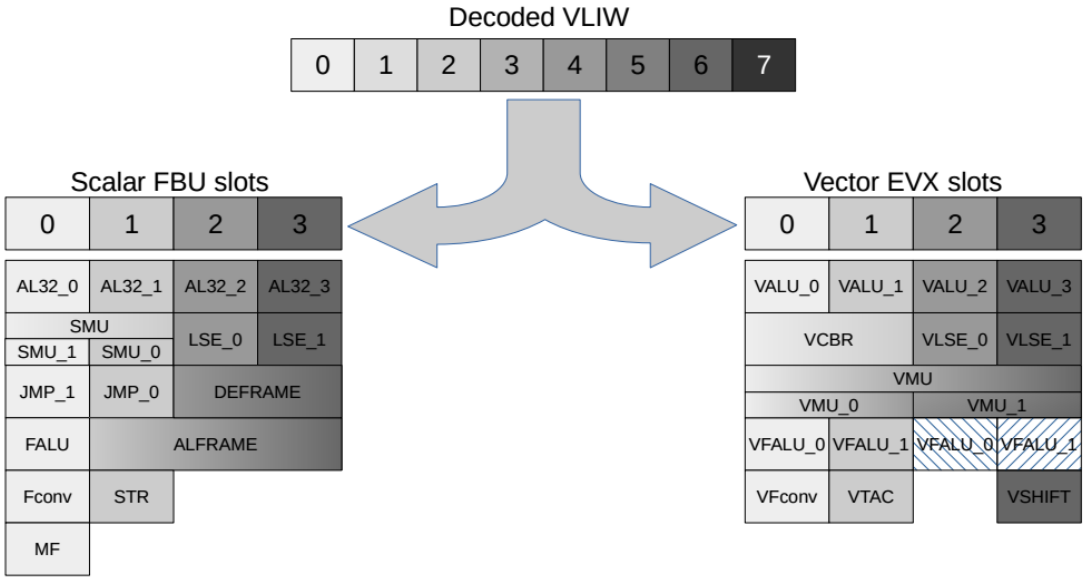


Рисунок 20.3. Вычислительные слоты DSP ядра Elcore50

Вычислительная часть DSP ядра состоит из двух частей: скалярная часть в FBU и векторная часть в EVX. В декодере VLIW в PDC формируется до 8-ми команд, из них до 4-ёх поступает в исполнительные слоты FBU и до 4-ёх в исполнительные слоты EVX. Сами исполнительные блоки состоят из следующих вычислителей (На рисунке постфиксами «_0», «_1» и т.д. помечены копии одного исполнительного блока):

- FBU – скалярный вычислительный блок. Помимо исполнительных блоков содержит регистровый файл (RF) из 32-ух регистров по 64 бита с 10-ю портами чтения и 6-ю портами записи и регистры предикатов (PRF) из 16 регистров по 1-му биту с 8-ю портами чтения и 4-мя портами записи.
    - AL32 – арифметико-логический вычислительный блок. Исполняет большой набор инструкций, а также используется для скалярных атомарных операций на внешней памяти и для некоторых других команд, на пример для ALFRAME. Один блок AL32 может использовать до 3-ёх портов скалярного RF и выполняет операции от 1-ого до 3-ёх тактов.
    - SMU (Scalar Multiplication Unit) – блок скалярного умножения. Выполняет умножения как над целочисленными операндами (знаковыми и беззнаковыми, комплексными и дробными форматами), так и над значениями с плавающей точкой. Так же выполняет составные умножения с накоплением или вычитанием. Размеры операндов и результатов: 32 бита (SMU_0/SMU_1) или 64 бита (SMU).
    - LSE – блок, выполняющий скалярные пересылки с памятью.
    - ALFRAME, DEFRAME – распределённая логика, выполняющая команды управления программным стэком ALFRAME, DEFRAME, DEFRAMER.

ALFRAME использует один AL32 и оба LSE. DEFRAME и DEFRAMER используют только два LSE.

- MF (MultiFunc) – блок, выполняющий различные математические функции, на пример вычисление косинуса/синуса, корня, экспоненты, логарифма или деление. Так же содержит два подблока fi_d0m, fi_d1m для первого и второго приближения обратной величины и корня обратной величины над 64-ёх битным значением в формате плавающей точки.
- JMP – блок выполнения переходов, частично входит в состав IFE. В одном VLIW может быть один переход или два взаимоисключающих по двум адресам (аппаратная реализация IF-ELSE).
- FALU – арифметико-логический вычислительный блок над числами в формате плавающей точки.
- Fconv – блок преобразования формата представления чисел.
- STR – блок, выполняющий байтовые операции (сравнение, копирование и поиск) над строками.

- EVX – векторный вычислительный блок. Состоит из 8-ми 64-ёх битных SIMD секций и отдельного межсекционного вычислителя. В каждой секции находятся векторный регистровый файл (VRF) из 64-ёх регистров по 64 бита с 8-ю портами чтения и 4-мя портами записи и векторные регистры предикатов (VPRF) из 16-ти регистров по 8 бит с 8-ю портами чтения и 4-мя портами записи, а так же блок аккумуляторов (VAC) из 16-ти регистров по 16x32 бит каждый.
  - VALU – векторный арифметико-логический вычислительный блок. Занимает от 1-ого до 2-ух слотов EVX.
  - VCBR – блок межсекционных вычислений. Выполняет такие операции, как упаковка, распаковка, вычисление гистограммы, редукция, перестановка. Занимает от 1-ого до 2-ух слотов EVX.
  - VLSE – блок, выполняющий векторные пересылки с памятью.
  - VMU – блок векторных умножений и фильтрации. Может выполнять действия как над векторными регистрами, так и над аккумуляторами. Занимает от 1-ого до 4-ёх слотов EVX.
  - VFALU – арифметико-логический вычислительный блок над векторами в формате плавающей точки. Штрихом помечены те же самые блоки, но использующие другие векторные слоты EVX. Так же может занять сразу два слота, за счёт VFALU_1.
  - VFconv – блок преобразования формата представления чисел в векторе. Может занять дополнительный 1-ый или 3-ий слот по записи.
  - VSHIFT – блок сдвига входного вектора.

- VTAC – блок обмена между векторным регистровым файлом и файлом аккумуляторов с дополнительными преобразованиями. Так же находится вне SIMD секций. Может занять дополнительный 0-ой слот по записи.

Векторные инструкции помимо EVX слотов так же могут занимать скалярные FBU слоты, если используют скалярные регистры RF. Подробное описание микроархитектуры DSP ядра Elcore50 приведено в документе «DSP-ядро Elcore50. Руководство пользователя». Перечень VLIW инструкций приведён в документе «DSP-ядро Elcore50. Система инструкций». Описание работы с блоком VMMU приведено в документе «DSP-ядро Elcore50. Руководство пользователя блока VMMU», на блок VDMA в документе «DSP-ядро Elcore50. Руководство пользователя блока VDMA», а на блок Event Ctrl. в документе «DSP-ядро Elcore50. Руководство пользователя контроллера EVENT_CTRL».

20.3Карта памяти

Таблица 20.2. Сопоставление базовых смещений

Именование базового смещения	Сопоставление системным смещениям	**Описание**
BASE0	DSP_LOCAL_XYRAM + PRAM 0x02E0_0000	Внутренний базовый адрес локальной памяти DSP
	DSP0_XYRAM + PRAM 0x0300_0000	Системный базовый адрес локальной памяти DSP0
	DSP1_XYRAM + PRAM 0x0320_0000	Системный базовый адрес локальной памяти DSP1
BASER	DSP_LOCAL_REGS 0x0280_0000	Внутренний базовый адрес регистров DSP
	DSP0 0x0198_0000	Системный базовый адрес регистров DSP0
	DSP1 0x01C8_0000	Системный базовый адрес регистров DSP1

Основные диапазоны памяти ядра DSP приведены в таблице 20.2. В ядре Elcore50 существуют два диапазона адресов, указывающих на одни и те же регистры или одну и ту же память. Первый такой диапазон является внутренним для DSP ядра, доступен только самому ядру, не зависит от номера ядра и зарезервирован на системном уровне. Второй

диапазон – системный и доступен любому мастеру. Обращения со стороны конкретного DSP ядра в свои адреса по любому диапазону не выходят за пределы Elcore50, за исключением PRAM (чтение и запись по внутреннему базовому адресу не доступно).

Таблица 20.3. Перечень используемых DSP диапазонов адресов

Именование базового смещения	Смещение	Размер, Кбайт	Описание
BASE0	0x000000	512	Память данных XYRAM, блочный режим
	0x080000	512	Память данных XYRAM, интерлив режим
	0x100000	64	Память программ PRAM
BASER	0x000000	16	Регистры ядра DSP Elcore50
	0x200000	4	Системные регистры DSP
	0x220000	4	Регистры контроллера ивентов
	0x230000	4	Регистры VDMA
	0x240000	4	Регистры VMMU
	0x250000	4	Регистры аппаратного трэйсера
	0x260000	20	Отладочные регистры L1CI
	0x2B0000	1	Отладочные регистры L0CI
	0x300000	20	Отладочные регистры L1CD
	0x350000	5	Отладочные регистры L0CD
	0x370000	36	Отладочные регистры L2C

20.4Матрица коммутации DSP и VDMA

Обращения со стороны ядра DSP к соседнему ядру DSP аналогично обращению к любым другим системным ресурсам, проходит через коммутатор Big Bandwidth Data interconnect.

Векторные обращения DSP разделены на векторно-блочные и векторно-индексные. Векторно-блочные – обращения по 64 байта не ограничены диапазонами адресов. Векторно-индексные – обращения по 8 независимым адресам от байта до 8 байт и доступные только в направлении XYRAM.

Таблица 20.4. Матрица коммутации DSP и VDMA

Ресурс	DSP, exec	DSP, scalar	DSP, vector	VDMA PTW
XYRAM память DSP, локальные адреса		+	+	
XYRAM память DSP, системные адреса	+	+	+	+
Регистры DSP, локальные адреса		+		
Регистры DSP, системные адреса	+	+	+	+
PRAM память DSP, локальные адреса	+			
PRAM память DSP, системные адреса	+	+	+	+
Регистры и память SDR подсистемы	+	+	+	+
DDR и остальные системные ресурсы	+	+	+	+

20.5 VMMU

20.5.1 Термины

PPN — Physical page number, часть физического адреса, указывающего на смещение страницы в памяти.

VPN — Virtual page number, часть виртуального адреса, используемого для трансляции. В процессе трансляции заменяется на соответствующий PPN.

PTE – Page Table Entry, запись в таблице страниц, 64-х разрядное слово в памяти, описывающее размер выделенной области, параметры трансляции и права доступа.

Leaf — PTE, которая является последней в цепочке записей и которая содержит описание прав доступа, части физического адреса и другие аттрибуты. Leaf не является указателем на следующий уровень и в поле PPN содержат адрес трансляции. Остальные PTE в PPN содержат ссылку на следующий уровень.

PTW – Page Table Walker, устройство способное считывать цепочки PTE из памяти по запросу из TLB, пока не достигнет Leaf или невалидную PTE.

TLB – Translation Lookaside Buffer, буфер преобразования адреса, проверяет права доступа и транслирует адреса по имеющимся во внутреннем кэше PTE.

20.5.2 Введение

VMMU является блоком управления доступа к памяти, построенным по архитектуре RISC-V *Priveleged Architecture v1.9draft*. Состоит из двух основных частей – TLB буфера и устройства PTW аппаратно обслуживающего промахи в связанных с ним TLB. Для использования в рамках DSP процессора, часть архитектуры DSP связанная с работой с памятью адаптирована к архитектуре RISC-V (введены необходимые регистры и сигналы для управления работой TLB).

В рамках проекта развивается два набора блоков, представляющих из себя VMMU: DSP версия и стандартная VDMA/QDMA/RISC версия. DSP версия отличается от VDMA/QDMA/RISC версии в основном использование собственного протокола шины, VDMA/QDMA/RISC версия использует AXIv3.

Блок VMMU является IOMMU и не интегрирован в ядра DSP/RISC или VDMA/QDMA. Т.е. нет тесной связи между транзакциями и событиями в TLB. Так же DSP/ RISC могут сами менять настройки TLB и в процессе работы, и по событию. Тем самым полной защиты нет. Так же TLB не связан с кэшами ядер, нет обратного поиска физического адреса в VT (virtually tagged) кэшах. Отсюда следует, что защиты от алиасинга нет, один и тот же физический адрес может содержаться в кэшах под двумя виртуальными адресами.

20.5.3 Устройство VMMU

Изначально написан на Chisel (scala синтаксис) для RISC-V и адаптирован для RISC/QDMA/DSP/VDMA в двух вариациях: с интерфейсом AXIv3 и AXI-подобная внутренняя шина DSP. Максимальное число TLB равно 8 и по 2 TLB на устройство. Максимум 4 обслуживаемых устройства.**Параметры имплементации**

- Обслуживаемых устройств DEV_NUM: 2
- Общее число TLB блоков TLB_NUM: 4. По 2 канала (запись и чтение) на устройство.
- Встроенный кэш Leaf PTE с количеством записей для каждого TLB: 32
- Встроенный кэш Non Leaf PTE с количеством записей для каждого PTW: 16
- Модель трансляции RISC-V: Sv48 (до 4-ёх уровней)
- Эффективная ширина физического адреса: 41 бит
- Ширина поля ASID: 7 бит
- Максимальная ширина виртуального адреса: 48 бит (12 бит индекс и по 9 бит на уровень)

20.5.3.2 Специфичные параметры для отдельных блоков/проектов

- TLB для VDMA использует полный 41-о битный виртуальный адрес, который указан в дескрипторе задания, без ASID
- TLB для DSP при включённом VMMU (выставленный Status_VM) использует 32-ух битный виртуальный адрес с конкатенацией ASID. В результате на каждый ASID используется своя таблица трансляции. Выключенный VMMU (сброшенный Status_VM) игнорирует ASID (использует 0 вместо него)
- TLB для RISC/QDMA использует 32-ух битный виртуальный адрес без ASID
- Доступ PTW в память: разделяет канал с QDMA/VDMA

20.5.3.3 Состав VMMU

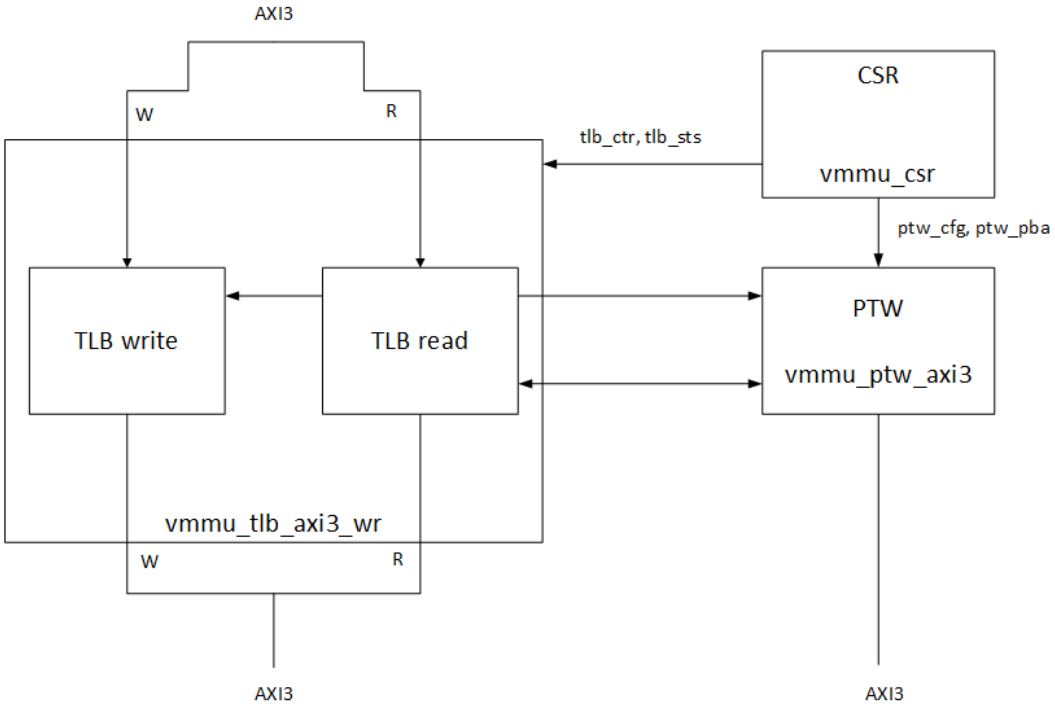


Рисунок 20.4. Подсистема трансляции

Каждый инстанс VMMU содержит один блок PTW, некоторое число блоков TLB, блок регистров CSR и соответствующие обвязки шин. Для RISC+QDMA и DSP+VDMA используется 4 блока TLB: TLB0 — порт записи RISC/DSP, TLB1 — порт чтения/исполнения RISC/DSP, TLB2 — порт записи QDMA/VDMA, TLB3 — порт чтения QDMA/VDMA.

PTW обслуживает все подключённые к нему TLB. CSR содержит регистры управления и прерывания. Логика шин находится вне блоков PTW и TLB, образуя обёртки над ними.

20.5.3.4 Регистровая модель VMMU

Таблица 20.5. Список регистров VMMU

Наименование	R/W	Бит	Адрес	Reset	Описание
PTW_PBA_L	RW	32	BASE+0x0	0x0	Физический адрес первого уровня таблицы преобразования адресов. Младшая часть
PTW_PBA_H	RW	9	BASE+0x4	0x0	Физический адрес первого уровня таблицы преобразования адресов. Старшая часть
PTW_CFG	RW	22	BASE+0x8	0x100000	Регистр настроек блока PTW и общих параметров всех TLB
TLBXCPT_NUM	RW	4	BASE+0xC	0x0	Регистр выбора номера TLB для считывания регистров TLBXCPT_ADDR и TLBXCPT_TYPE. 0 бит выбирает канал записи(1) или чтения. [3-1] биты выбирают номер TLB. Для выбора TLB0: 0x1 TLB1: 0x0 TLB2: 0x3 TLB3: 0x2 При выборе несуществующего TLB произойдёт запись/чтение 0-ого.
TLBXCPT_ADDR	RO	32	BASE+0x10	0x0	Адрес, обращение по которому вызвало исключение/прерывание по нарушению прав доступа
TLBXCPT_TYPE	RO	4	BASE+0x14	0x0	Тип события (исключение или прерывание, в зависимости от подключения в проекте). 0x0 – нет события; 0x1 – невалидная страница PTE; 0x2 – запись в область без прав записи; 0x3 – чтение из области без прав чтения; 0x4 – исполнение инструкции из области без прав исполнения; 0x5 – 4’hF – зарезервировано
MAPSEG_START_L	RW	32	BASE+0x18	0x0	Начальный адрес диапазона памяти без преобразования адресов. Младшая часть
MAPSEG_START_H	RW	9	BASE+0x1C	0x0	Начальный адрес диапазона памяти без преобразования адресов. Старшая часть
MAPSEG_END_L	RW	32	BASE+0x20	0xffffffff	Конечный адрес диапазона памяти без преобразования адресов. Младшая часть
MAPSEG_END_H	RW	9	BASE+0x24	0x1ff	Конечный адрес диапазона памяти без

					преобразования адресов. Старшая часть
MAPSEG_ENABLE	RW	1	BASE+0x28	0x0	Включение проверки виртуального адреса на предмет попадания в диапазон памяти без преобразования адресов.
TLB_CTRL	RW	7	BASE+0x40 +0x4* DEV_NUM	0x7	Регистры управления работой блоками TLB, для одного устройства DEV_NUM

Таблица 20.6. Регистр PTW_CFG

Наименование	R/W	Разряд	Reset	Описание
Invalidate	W1	0	0x0	Инвалидация всех связанных с PTW кэшей таблиц в TLB
Status_VM	RW	5:1	0x0	Включение трансляции адресов и проверки прав доступа. 3-ий бит выставлен – включение трансляции и защиты (0x8, 0xA и т.д.); 0x1 — режим Base & Bound (не реализован, но принудительно включает буфера на шине); Остальные значения – пропуск трансляции и защиты;
Status_MPRV	RW	6	0x0	Включение использования PRV1 при проверке прав доступа вне зависимости от AWPROT[0]/ARPROT[0]. Не работает для потока исполнения инструкций
Status_PRV1	RW	8:7	0x0	Настройка уровня доступа. 0x0 – User; 0x1 – Supervisor; Остальные значения отключают трансляцию и защиту
Status_PRV	RW	10:9	0x0	
Status_IRQ	RO	14:11	0x0	Биты событий от соответствущих TLB [0] бит — событие от TLB0; [1] бит — событие от TLB1; [2] бит — событие от TLB2; [3] бит — событие от TLB3;
A_Cache	RW	18:15	0x0	Атрибуты ARCACHE/AWCACHE шины AXI, передаваемые при запросах записей PTE в памяти блоком PTW
A_Prot	RW	21:19	0x2	Атрибуты ARPROT/AWPROT шины AXI, передаваемые при запросах записей PTE в памяти блоком PTW
Prefetch	RW	22	0x0	Загружать следующий PTE отдельным запросом
Fetchtwo	RW	23	0x0	Загружать два PTE за раз в режиме AXI burst без разбивки на границе 4К.

Таблица 20.7. Регистр TLB_CTRL

Наименование	R/W	Разряд	Reset	Описание
passthrough_TLB_write_channel	RW	0	0x1	Отключение преобразования адреса для канала записи
passthrough_TLB_read_channel	RW	1	0x1	Отключение преобразования адреса для канала чтения
passthrough_TLB_exec_channel	RW	2	0x1	Отключение преобразования адреса для канала исполнения
xpct_retry_TLB_write_channel	W1	3	0x0	Повторить транзакцию записи
xpct_retry_TLB_read_channel	W1	4	0x0	Повторить транзакцию чтения
xpct_dummy_TLB_write_channel	W1	5	0x0	Сбросить транзакцию записи, вызвавшую исключение/прерывание
xpct_dummy_TLB_read_channel	W1	6	0x0	Вернуть 0x0 на транзакцию чтения, вызвавшую исключение/прерывание
xpct_autodummy_TLB_write_channel	RW	7	0x0	Автоматически сбрасывать транзакции записи, вызвавшие исключение или прерывание
xpct_autodummy_TLB_read_channel	RW	8	0x0	Автоматически возвращать 0x0 на транзакции записи, вызвавшие исключение или прерывание

20.5.3.5 Порядок трансляции адреса в VMMU

Порядок инициализации такой:

1) Создать структуру таблиц PTE в памяти. Формат указан в таблице Таблица 20.8.

2) Указать адрес начала первого уровня таблиц PTE в регистрах PTW_PBA_L и PTW_PBA_H

3) Включить трансляцию, выставив в регистре Status_VM значение 0xA.

4) Отключить пропуск трансляции каналов записи/чтения/исполнения в регистре TLB_CTRL определённого блока TLB.

5) Проверить, что нетранслируемый диапазон MAPSEG не используется: регистр MAPSEG_ENABLE сброшен.

6) Так же включить пропуск трансляции может шина AXI через ARPROT[1]/AWPROT[1]. Для AXI-подобной шины DSP так же присутствуют эти сигналы и они напрямую берутся с регистра PCU[12].MREGIONS.

7) Значения в регистрах PRV/PRV1 выше 1 (Supervisor) были зарезервированы для прав доступа Hypervisor и Machine. Выставление этих значений выключить трансляцию адресов и сверку прав доступа. PTE считываться будут.

Пункты 4, 5 и 6 не влияют на проверку прав доступа. Страницы PTE будут считаны из памяти, проверены на валидность и сверены с правами доступа в PRV/PRV1.

Таблица 20.8. Формат записи PTE в памяти

Наименование	Разряды	Описание
V	0	Признак действительности записи PTE
Type	4:1	Указатель на следующий уровень трансляции или кодировка параметров доступа (Leaf).
R	5	Признак Referenced
D	6	Признак Dirty
UserDefined	9:7	Reserved for Software
PPN[0]	18:10	Части физического адреса
PPN[1]	27:19	
PPN[2]	36:28	
PPN[3]	45:37	
RSVD	63:46	Зарезервировано

Изначально в описании VMMU в RISC-V был бит Global, но в текущей реализации он не поддерживается и потому не включён в таблицы.

Таблица 20.9. Формат поля Type в PTE

Type	Supervisor			User		
	R	W	X	R	W	X
0 1	Указатель на следующий уровень					
2	●			●		●
3	●	●		●	●	●
4	●			●		

Type	Supervisor			User		
	R	W	X	R	W	X
5	●	●		●	●	
6	●		●	●		●
7	●	●	●	●	●	●
8	●					
9	●	●				
10	●		●			
11	●	●	●			
12	●					
13	●	●				
14	●		●			
15	●	●	●			

Можно изначально не задавать всю структуру таблиц PTE, а по мере необходимости. В этом случае 1-ый пункт опускается. Теперь появляется необходимость реагировать на события от TLB. Тип события от блоков TLB определяется проектом, в котором используется VMMU. Для DSP события заведены на прерывания. В обработчике надо сделать следующие действия:

1) Убедится, что событие было от блока TLB. В случае DSP из IRQR сразу будет известно какой канал вызвал прерывание: на запись или на чтение/исполнение, либо TLB обслуживающий VDMA. Эта же информация доступна в регистре Status_IRQs

2) Записываем в регистр TLBXCPT_NUM номер TLB из предыдущего пункта

3) Из регистров TLBXCPT_TYPE и TLBXCPT_ADDR определяем причину и адрес срабатывания события.

4) Если страницы нет, то заводим, а если недостаточны права доступа – меняем PTE

5) В регистре PTW_CFG выставляем Invalidate для сброса кэшей во всех TLB

6) Доступно либо повторить транзакцию, либо сбросить:

- Повторяем транзакцию с помощью xpct_retry_TLB_read_channel для чтения и xpct_retry_TLB_write_channel для записи
- Если вместо повторения надо сбросить неверное обращение, то используем поля xpct_dummy_TLB_write_channel для сброса записи в память и xpct_dummy_TLB_read_channel для создания фиктивного ответа на чтение.

Пункт 6 сбрасывает событие, но в случае DSP необходимо так же сбросить IRQR/DQSTR.

Существует 4-е возможных размера страниц, описываемых с помощью PTE:

- 512 Гбайт — используется один уровень и одна запись PTE на страницу: **Рисунок 20.8**
- 1 Гбайт — используется два уровня и две записи PTE на страницу: **Рисунок 20.7**
- 2 Мбайт — используется три уровня и три записи PTE на страницу: **Рисунок 20.6**
- 4 Кбайт — используется четыре уровня и по четыре записи PTE на страницу: **Рисунок 20.5**

Виртуальный адрес разбивается на 5-ть компонент: индекс 12 бит (PAGE OFFSET) и четыре уровня (VPN0, VPN1, VPN2, VPN3) по 9 бит каждый. Общее число доступных страниц определяется используемой шириной виртуального адреса. Для DSP это 32бита на адрес и 7 бит на ASID, который конкатенируется над адресом. Отсюда следует что для VPN3 не остаётся места и доступна всего одна страница 512 Гбайт, и никаких других страниц создано быть не может, так как эта же страница будет иметь Type отличный от 0 или 1. Т.е. для виртуального адреса в 39 бит существует только одна Leaf страница 512Гбайт.

Порядок определения размера страницы такой:

1) Читается первая запись PTE по адресу из регистров PTW_PBA_L и PTW_PBA_H
2) Если она валидна, то проверяется Type. В противном случае выставляется событие невалидной PTE записи
3) Если Type не равен 0 или 1, то страница является Leaf и имеет размер 512 Гбайт. Физический адрес получается заменой VPN3 на PPN3, сложением остальных частей VPN с соответствующими PPN и добавлением индекса из виртуального адреса : {PPN3, PPN2+VNP2,PPN1+VNP1,PPN0+VNP0,PAGE OFFSET}.
4) Если Type равен 0 или 1, то PTW читает PTE следующего уровня по адресу {PPN3,PPN2,PPN1,PPN0,VPN2}. Где все PPN* берутся из уже прочитанной в пункте 1 PTE
5) Если Type 2-ой PTE не равен 0 или 1, то страница является Leaf и имеет размер 1 Гбайт. Физический адрес получается заменой VPN3 на PPN3 и VPN2 на PPN2,

сложением остальных частей VPN с соответствующими PPN и добавлением индекса ищ виртуального адреса: {PPN3, PPN2, PPN1+VNP1, PPN0+VNP0,PAGE OFFSET}.

6) Если Type 2-ой PTE равен 0 или 1, то PTW читает PTE 3-его уровня по адресу {PPN3,PPN2,PPN1,PPN0,VPN1}

7) Так повторяется до тех пор, пока не будет найден Leaf, либо запись PTE будет не валидна, либо число уровней превысит 4-е.

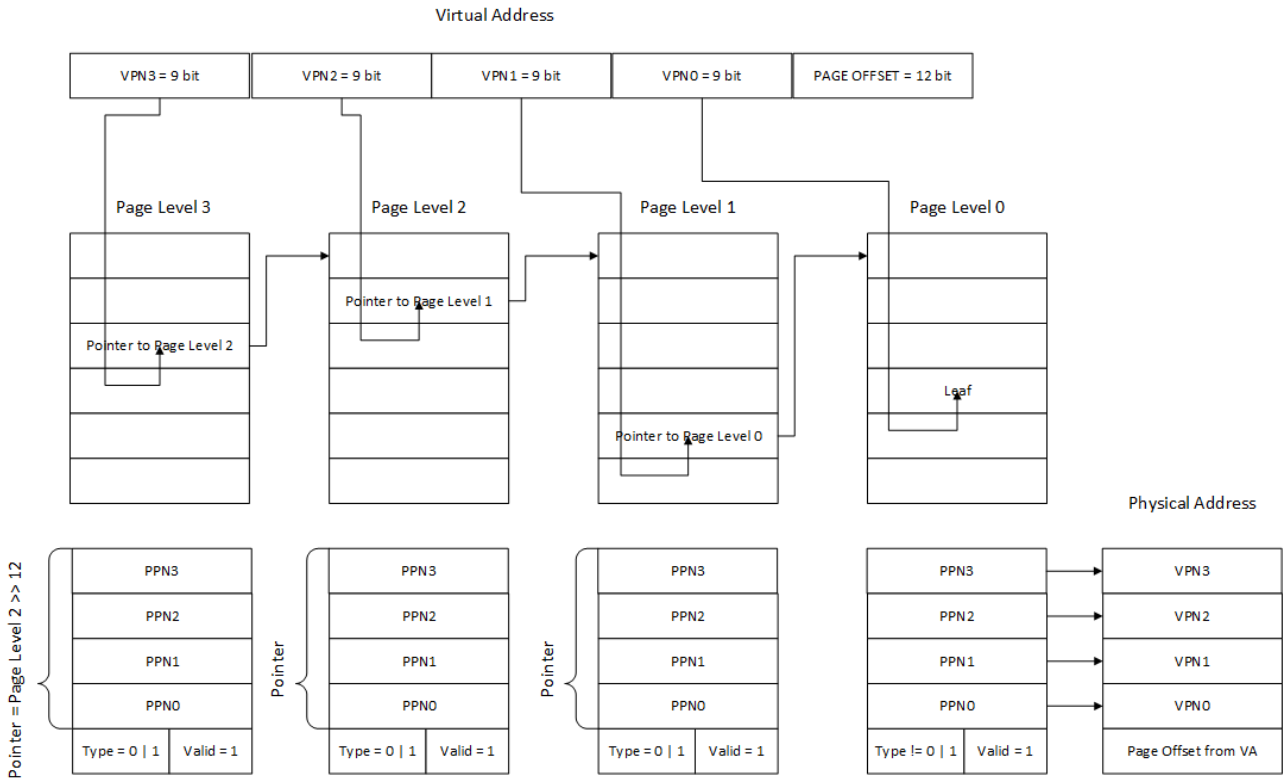


Рисунок 20.5. Принцип трансляции 4КБ страницы

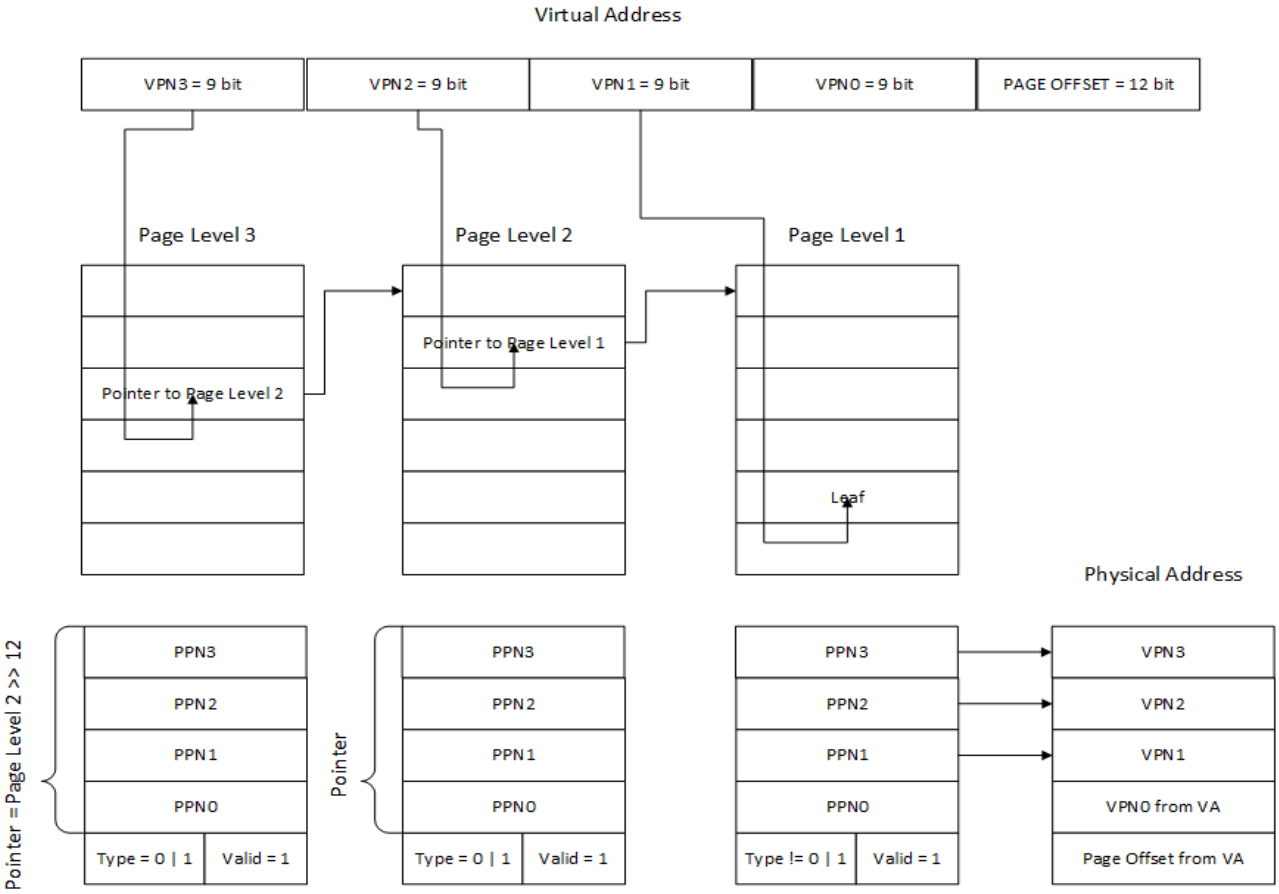


Рисунок 20.6. Принцип трансляции 2МБ страницы

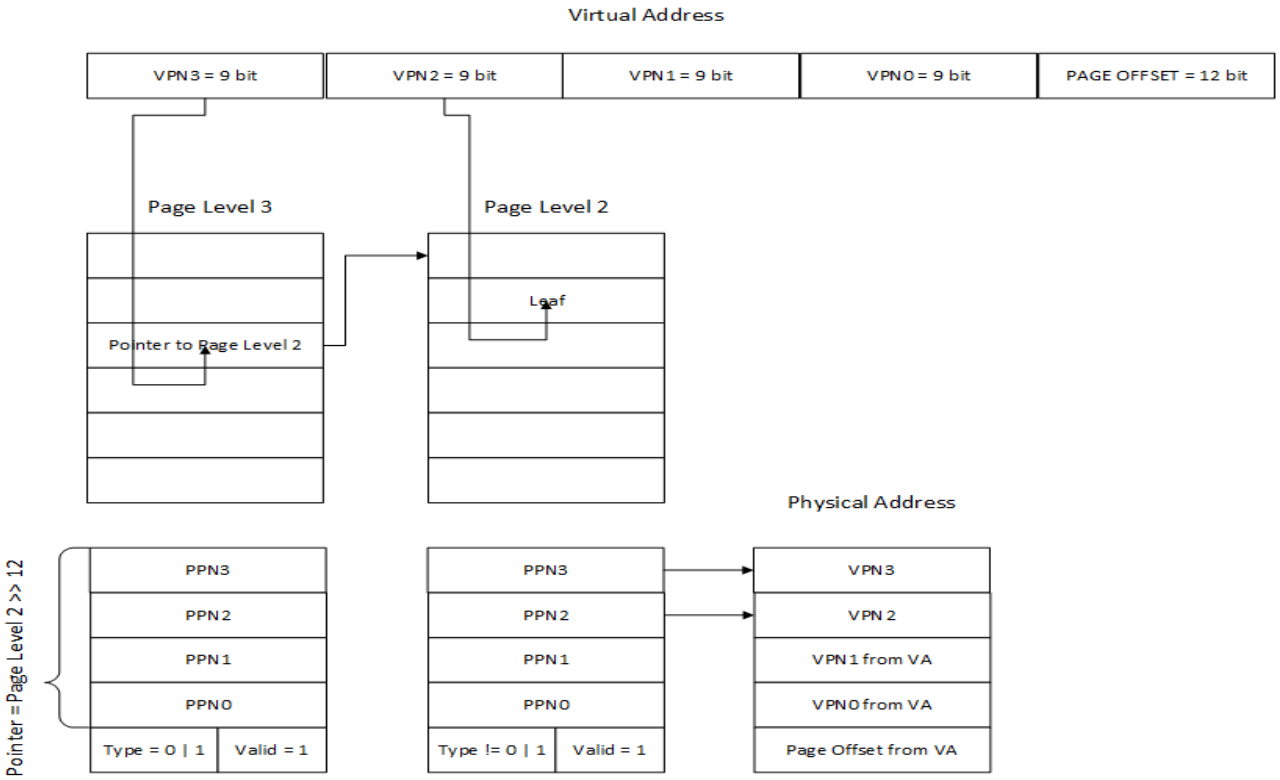


Рисунок 20.7. Принцип трансляции 1ГБ страницы

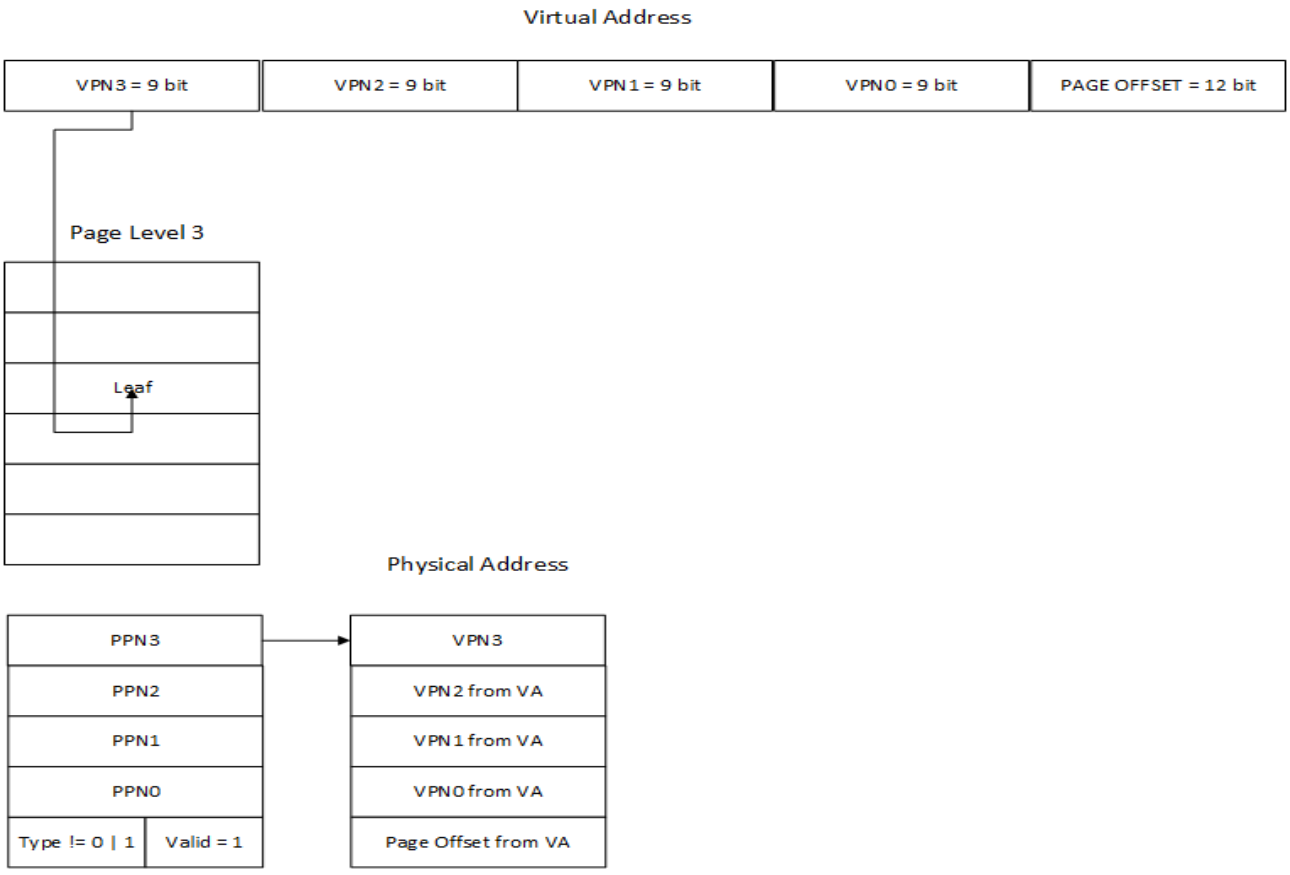


Рисунок 20.8. Принцип трансляции 512ГБ страницы

20.5.3.6 Порядок ограничения прав доступа в VMMU

После включения VMMU и задания таблицы трансляции в памяти, VMMU будет так же проверять разрешено ли обращение в страницу памяти. Порядок проверки следующий:

1. Если пришла транзакция исполнения инструкции или не стоит бит MPRV, то права доступа в PTE будут сравниваться с регистром PRV.

2. Если транзакция чтения/записи и стоит бит MPRV либо транзакция чтения/записи сопровождается выставленным AWPROT[0]/ARPRTOT[0], то выбирается PRV1. В случае DSP AWPROT[0]/ARPRTOT[0] исходят из регистра PCU[12].MREGIONS.

3. Если полученные права из PRV/PRV1 не равны User или Supervisor, то трансляции адреса не будет и права доступа не будут сверены.

4. В зависимости от того, User права или Supervisor, из поля Type Leaf будет выбраны права доступа. В случае недостаточных прав будет выставлено событие. Транзакция останется висеть в ожидании таких же действий, как для отсутствующей таблицы трансляции в памяти.

20.5.3.7 Особенности реализации VMMU

1. В текущей реализации VMMU один регистр PTW_CFG для всех подключённых TLB. На примере VMMU для DSP, он управляет некоторыми аспектами и для DMA, и для DSP.
   В регистр PTW_CFG помимо инвалидации и разрядности адреса входит права доступа для процесса. Т.е. у VDMA/QDMA и DSP/RISC будут одни и те же права доступа (поля PRV/PRV1). Инвалидация сбрасывает кэши во всех TLB сразу.

2. В VMMU для QDMA не предусмотрено сигнала исполнения инструкции. Т.е. все чтения помечены как чтения данных. Все транзакции от RISC помечены как исполнение инструкций. В DSP версии VMMU сигнал присутствует и разделяется на данные и инструкции согласно обращению.

3. Нельзя с 100% вероятностью сказать, что в памяти находится валидная страница PTE. В реальном оборудовании может быть такая ситуация, что в бит валидности попала 1. При обращении PTW её считает и решит, что там что-то есть валидное. Никакой дополнительной проверки нет. При использовании VMMU надо обнулять память под PTE.

4. Переход в Dummy сброс осуществляется только если выставлено прерывание, т.е. при ошибочной записи в управляющий регистр заблокировать порт AXI нельзя. Повтор транзакции не может по стандарту AXI заблокировать порт.

5. В обоих окружениях RISC-QDMA-VMMU и DSP-VDMA-VMMU блок PTW делит выходной порт с QDMA/VDMA, имея при этом больший приоритет. Так как поток от DMA в разы больше, то PTW будет чаще ждать своей очереди в случае исчерпания outstanding транзакций.

6. Биты D (Dirty) и R (Referenced) обновляются в Leaf автоматически. Модификации атомарны только для порта VMMU. После записи обновлённого Leaf он будет прочитан заново. Остальные мастера не атомарны записям. Так же запись в страницу со сброшенным Dirty битом приводит к дополнительным задержкам (необходимо записать и считать результирующую Leaf PTE).

7. Изначально поле Status_VM содержало несколько режимов работы и основными были 0x1 и 0xA. Но в реализации оказалось, что помимо 0xA включают трансляцию несколько значений, у которых 3-ий бит равен 1, и между ними нет разницы.

8. Всего для всех имплементаций (RISC-QDMA-VMMU и DSP-VDMA-VMMU) подключено DEV_NUM=2 устройства и TLB_NUM=4 TLB. В CSR TLB_CTRL для каждого устройства. При чтении TLB_CTRL для отсутствующих устройств (3-ий или выше) читается 0-ой.

9. После записи или чтения от DSP может произойти активация префетча и в одну страницу произойдёт сразу несколько обращений. В случае, если первичное обращение надо сбросить без заведения PTE, надо сбросить ещё и все обращения со стороны префетча.

10. Механизм предвыборки fetchtwo неверно работает вместе с prefetch. Захватывает VPN из входного запроса, а не от prefetch. В результате сохраняет с другим отображением.

11. Механизм предвыборки fetchtwo выставляет burst запрос, но не проверяет 4К границу, а механизм prefetch выставляет fixed запрос.

21. БЛОК ЦИФРОВОЙ ОБРАБОТКИ РАДИО (DFE)

21.1 Общая информация

21.1.1 Назначение блока DFE

Блок DFE предназначен для построения цифровых программно-определяемых приемопередающих трактов радиостанций и модемов.

Блок DFE обеспечивает выполнение всех основных сложных функций по канальной обработке оцифрованных радиосигналов, таких как:

- многоканальные интерфейсы высокоскоростных современных ИМС ЦАП и АЦП;
- точные частотные и фазовые преобразования оцифрованных сигналов;
- цифровая канальная фильтрация;
- цифровая компенсация искажений в аналоговых входных и выходных цепях радиотракта;
- ресэмплирование и интерполяция;
- управление динамическим диапазоном принимаемых сигналов;
- определение параметров принимаемых сигналов;
- управление синтезаторами частоты, аттенюаторами, коммутаторами и другими ИМС радиотракта;
- высокопроизводительный многоканальный обмен данными приема-передачи с блоком цифровых сигнальных процессоров и аппаратных акселераторов.

21.1.2 Структурная схема

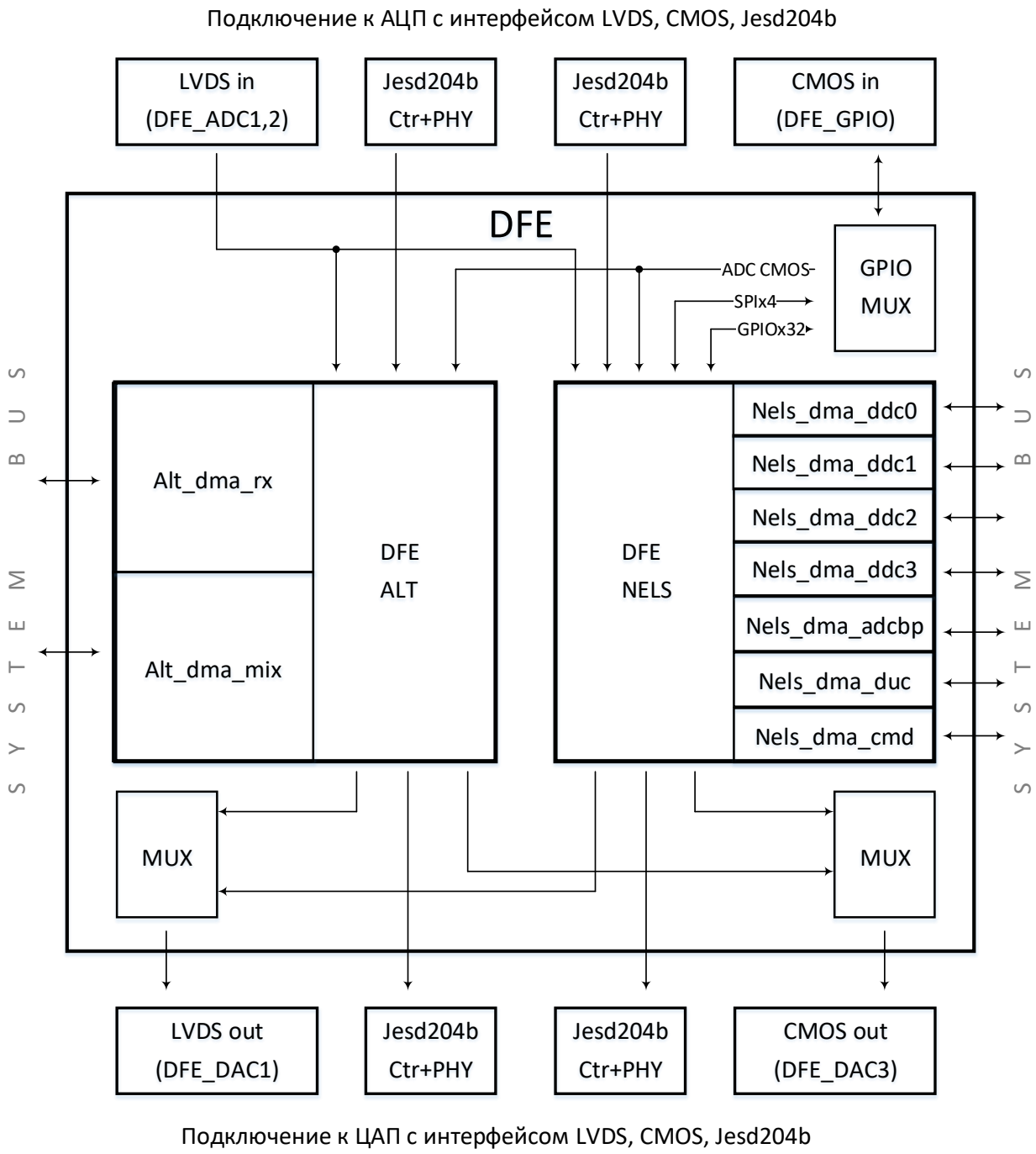


Рисунок 21.1. Структурная схема блока DFE

21.1.3 Блок DFE_NELS

21.1.3.1 Ключевые особенности блока

Блок DFE обеспечивает:

- Подключение до 3-х ИМС высокоскоростных АЦП с суммарным количеством принимаемых оцифрованных каналов до 12-ти. В том числе:
    - до 2-х 4-хканальных АЦП по интерфейсу LVDS;
    - один 4-хканальный АЦП по интерфейсу JESD204B.
- Подключение до 3-х ИМС высокоскоростных ЦАП с суммарным количеством формируемых передаваемых каналов до 8-ми. В том числе:
    - один 2-хканальны ЦАП по интерфейсу LVDS;
    - один 4-хканальный ЦАП по интерфейсу JESD204B;
    - один 2-хканальный ЦАП по интерфейсу CMOS.
- Разрядность подключаемых ЦАП и АЦП – до 16-ти.
- Максимальная частота сэмплирования данных, принимаемых от АЦП и передаваемых в ЦАП – до 500 МГц.
- Возможность использования различных тактовых частот для всех подключенных ЦАП и АЦП, в допустимых пределах.
- Возможность работы интерфейсов АЦП и ЦАП в режиме SDR и DDR.
- Цифровой прием и обработка сигналов до 16-ти реальных частотных каналов на промежуточной и радиочастоте, или до 8-ми комплексных каналов на промежуточной частоте одновременно.
- Формирование до 4-х передаваемых реальных или комплексных каналов на различных промежуточных или радиочастотах одновременно.
- Возможность формирования до 4-х смежных широкополосных каналов, суммированных в одном передаваемом частотном канале.
- Программируемая ширина принимаемых и формируемых каналов в пределах от десятков Гц до десятков МГц.
- Возможность передачи и приема сигналов с аппаратной поддержкой режимов ППРЧ и ЛЧМ.
- Высокоэффективные режимы снижения пик-фактора и линеаризации до 4-х передающих трактов при передаче сигналов со сложными видами модуляции.
- Эффективный режим управления динамическим диапазоном приема посредством адаптивного управления переключаемыми аттенюаторами радиотрактов.

- Цифровая компенсация смещения нулей, амплитуд и фаз квадратурных каналов в аналоговых цепях приемных и передающих трактов.

- Высокоэффективное гибкое управление всеми модулями блоков цифрового приема и передачи посредством 4-х канального автомата управления – DFE Commander, максимально разгружающего процессорные ядра микросхемы.

21.1.3.2 Структурная схема

21.1.3.2.1 Система управления

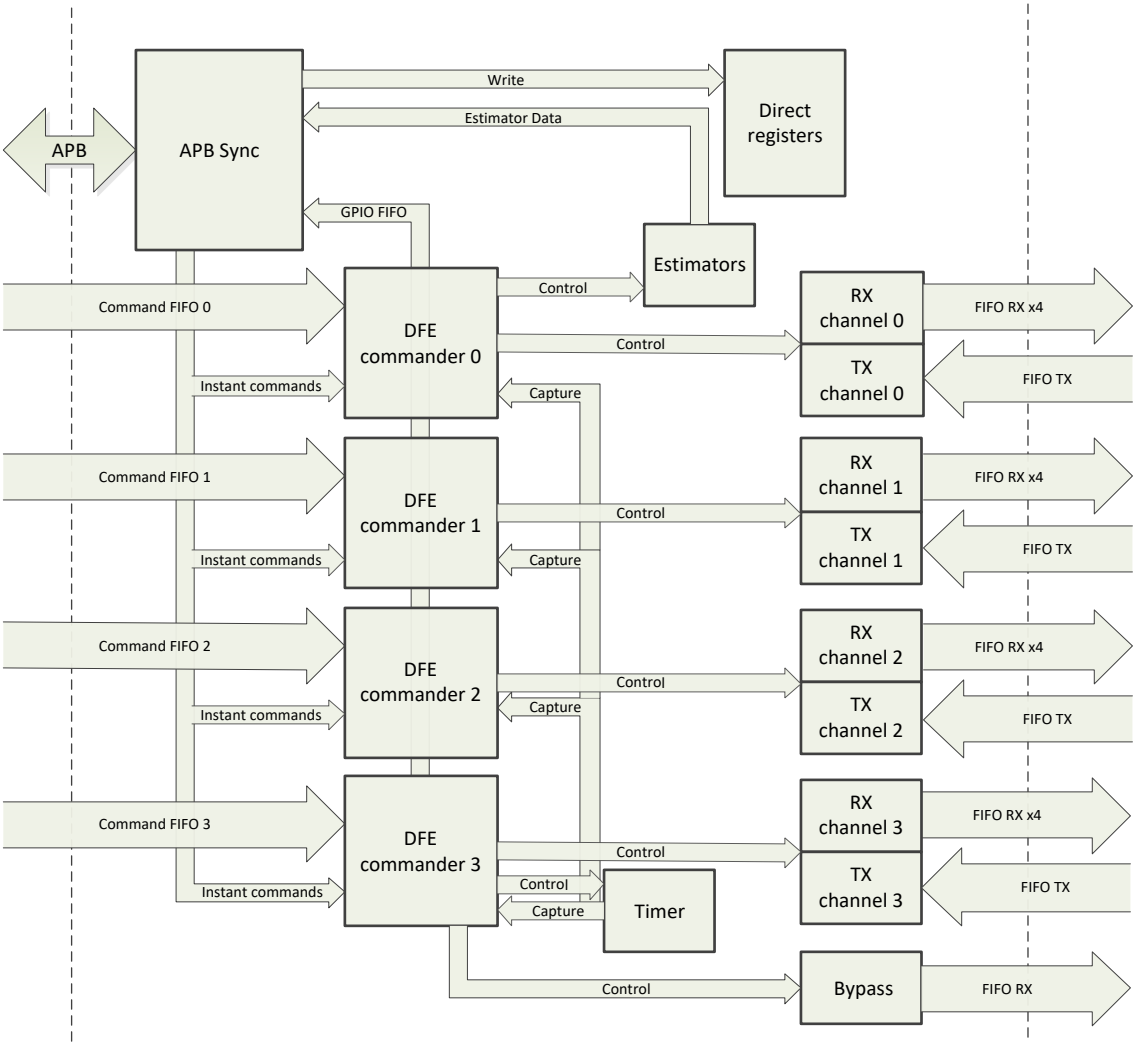


Рисунок 21.2. Схема системы управления DFE_NELS

Основное управление DFE осуществляют блоки DFE Commander. Каждый блок отвечает за управления четырьмя каналами приёмника и одним каналом передатчика. Дополнительно блок DFE Commander 3 отвечает за управление каналом bypass.

DFE Commander принимает команды с интерфейса ABP. Кроме того, возможна загрузка команд через Command FIFO для выполнения с точной привязкой ко времени.

Кроме того, через интерфейс APB идёт управление эстиматорами и общими регистрами.

21.1.3.2.2 Приёмник

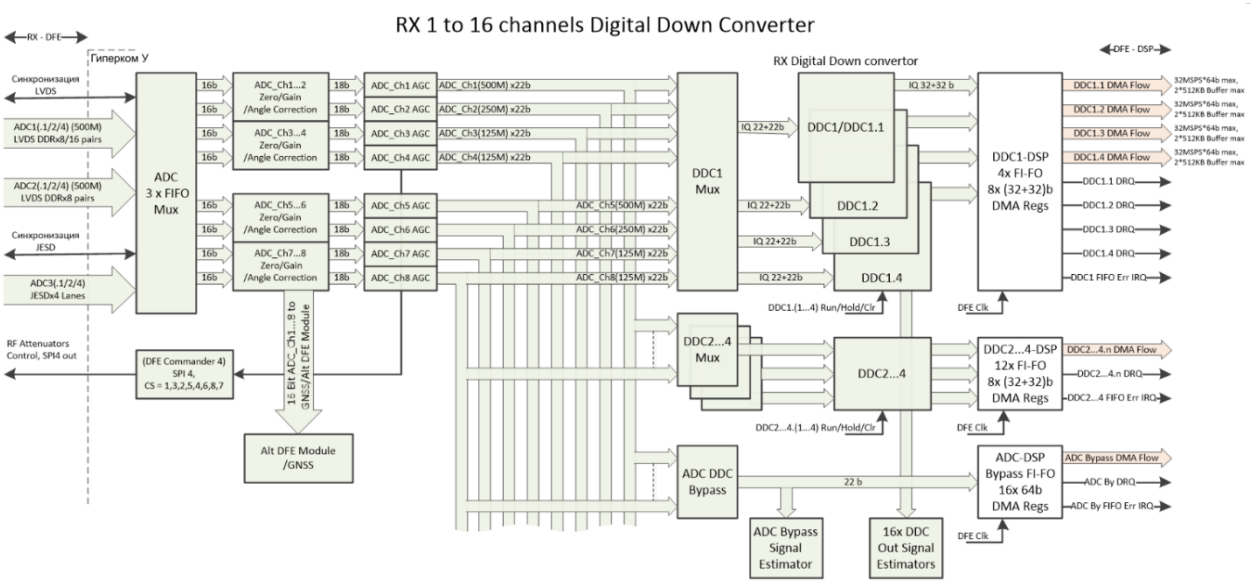


Рисунок 21.3. Схема приемника DFE_NELS

Модуль 16-ти канального цифрового приемного тракта RX 1 to 16 channels Digital Down Converter состоит из следующих основных модулей:

- модуль ADC 3xFIFO Mux интерфейсов подключения и мультиплексирования данных от различных типов подключенных ИМС АЦП;
- модули ADC Zero/Gain/Angle Correction, корректирующие неидеальность параметров принимаемых квадратурных сигналов;
- модули ADC_ChN AGC автоматической подстройки усиления аналоговых приемных трактов для расширения динамического диапазона принимаемых сигналов;
- модули DDC1-4 Mux мультиплексоров входных потоков принимаемых сигналов для дальнейшей обработки в модулях DDC;
- 16 модулей DDC, сгруппированных по 4 подмодуля – цифровых канальных конверторов и обработчиков принимаемых сигналов;
- выходные модули FIFO для организации очереди передачи синхронных данных из DDC асинхронно в оперативную память DSP-кластера;
- модуля ADC_DDC_Bypass, предназначенного для организации считывания данных от АЦП непосредственно в память кластера DSP без предварительной обработки.

21.1.3.2.3 Передатчик

-

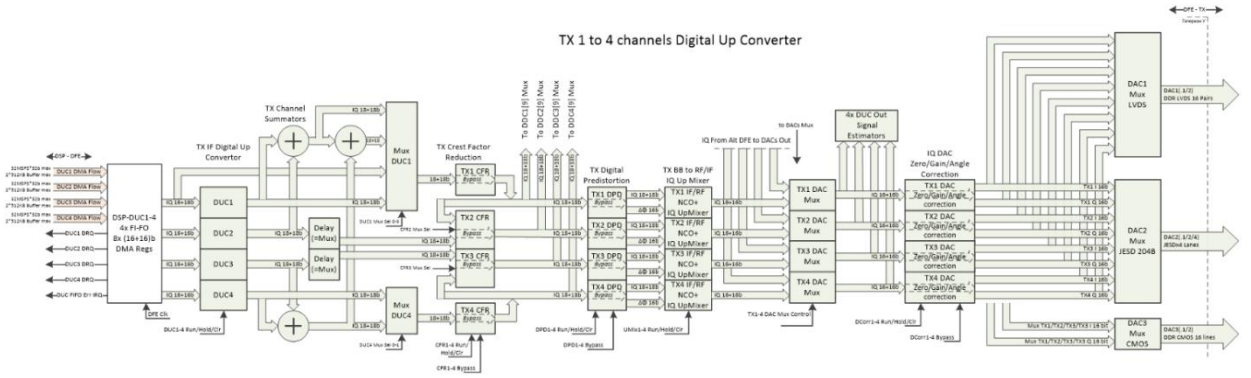


Рисунок 21.4. Схема передатчика DFE_NELS

Модуль TX 1 to 4 channels Digital Up Converter, структурная схема которого приведена в приложении Г, состоит из следующих основных узлов:

- четыре буфера FIFO DSP-DUC1-4, обеспечивающих передачу асинхронных потоков данных DMA от DSP синхронно к блоку DFE;
- цифровые повышающие конверторы DUC1-4, обеспечивающие интерполяцию, ресэмплирование, цифровую фильтрацию и перенос спектра каналов на промежуточные частоты для дальнейшего их суммирования-объединения с другими каналами;
- сумматоры «TX Channel Summators» с последующим мультиплексированием для формирования многоканальных передаваемых сигналов в одном канале ПРД;
- модули коррекции пик-фактора сигналов CFR1-4, обеспечивающие снижение пик-фактора сложных широкополосных сигналов до приемлемого уровня, обеспечивающего необходимый уровень средней мощности передатчика;
- модули дополнительной интерполяции «TX UpSampling», повышающие частоту дискретизации передаваемых сигналов до частоты работы ЦАП;
- модули введения адаптивных предыскажений «TX Digital Predistortion», обеспечивающие введение табличных предыскажений в формируемый сигнал с целью компенсации нелинейностей аналоговых усилителей мощности передатчиков;
- модули переноса частотного спектра каналов ПРД на заданные частоты, с возможностью установления начальной фазы, функциями ППРЧ и ЛЧМ;
- модули «IQ DAC Equalaser Zerro/Gain/Angle Correction» для коррекции смещения нулей, выравнивания усиления, устранения перекоса АЧХ и фаз квадратурных сигналов, возникающего в последующих квадратурных ЦАП и цепях аналоговой канальной фильтрации квадратурных сигналов;

- мультиплексоры для переключения потоков данных передаваемых каналов в различные ИМС ЦАП, в качестве которых могут быть использованы как квадратурные ЦАП, так и ЦАП для формирования каналов на промежуточной или передаваемой частоте.

21.1.4 Блок DFE_ALT

21.1.4.1 Ключевые особенности блока DFE_ALT

Блок цифровой обработки радио предназначен для построения приемных и передающих трактов систем радиосвязи и радиолокации. Блок имеет в своем составе четыре приемных (RX) и два передающих (TX) тракта. В блоке реализованы:

- функции преобразования входного сигнала с промежуточной частоты на низкую частоту с последующей фильтрацией и децимацией сигнала в RX тракте;
- функция согласованной фильтрации сигнала в RX тракте;
- весовое суммирование принимаемых сигналов в RX тракте;
- фильтрация, интерполяция и перенос частоты сигнала в TX тракте;
- модуляция, расширение спектра в TX тракте,
- синхронизация приемного и передающего трактов.

Обмен данными с АЦП и ЦАП осуществляются по 16 битным шинам CMOS и LVDS и JESD204B интерфейсу. Выделенные параллельные CMOS и LVDS АЦП и ЦАП интерфейсы поддерживают множество форматов обмена данными и обеспечивают реализацию до четырех каналов. Частота дискретизации сигналов, принимаемым и передаваемым по CMOS интерфейсам ограничена до 150 MHz, по LVDS интерфейсам – до 500 MHz.

21.1.4.2 Структурная схема

Блок- схема DFE приведена на Рисунок 21.5

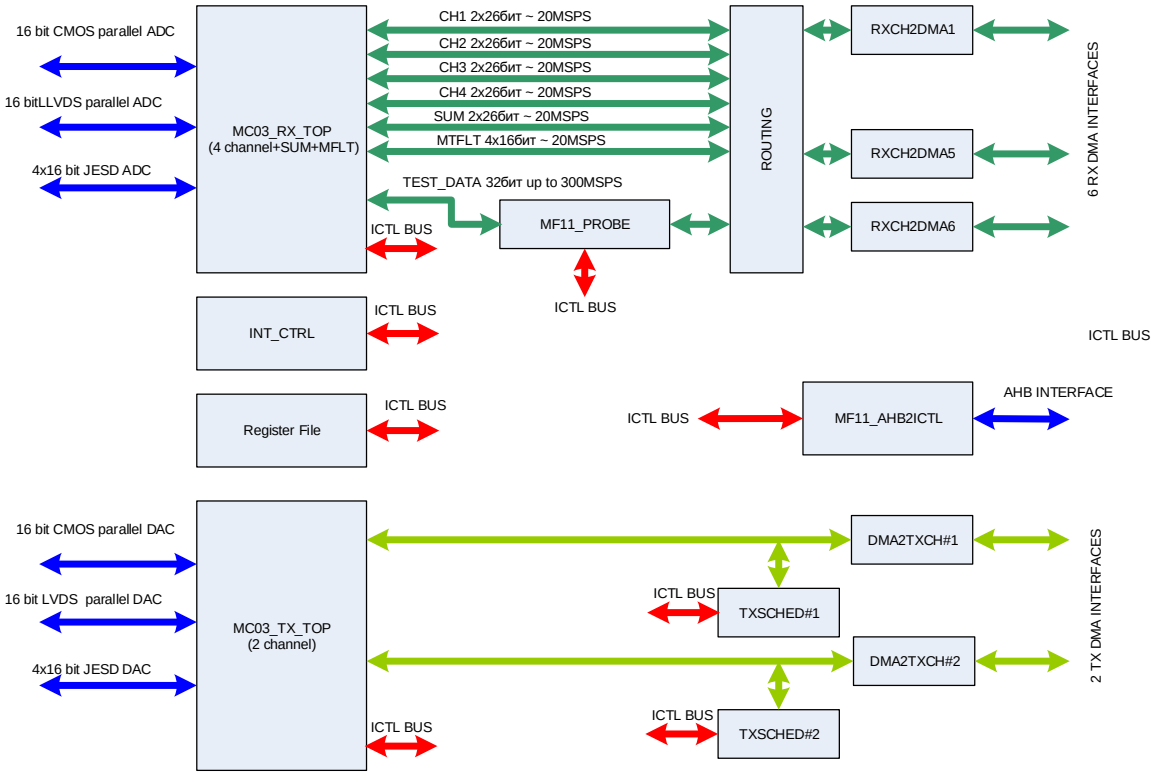


Рисунок 21.5. Функциональная схема DFE_ALT

Блок имеет внутренний интерфейс с шиной APB для регистрового доступа, предназначенный для его конфигурирования.

В приемной части DFE_ALT имеет шесть интерфейсов DMA с интегрированными FIFO для передачи выходных данных по AXI шине под управлением контроллера DMA. Интерфейсы могут подключаться к различным источникам сигнала:

- выходам каналов обработки DDC,
- выходному весовому сумматору,
- согласованному фильтру,
- источникам тестовых сигналов.

В передающей части DFE_ALT имеет два интерфейса DMA с интегрированными FIFO для приема данных по AXI под управлением контроллера DMA. Данные из интерфейсов подаются непосредственно на Tx каналы.

21.1.4.2.1 Приемная часть

Функциональная схема RX части представлена на Рисунок 21.6

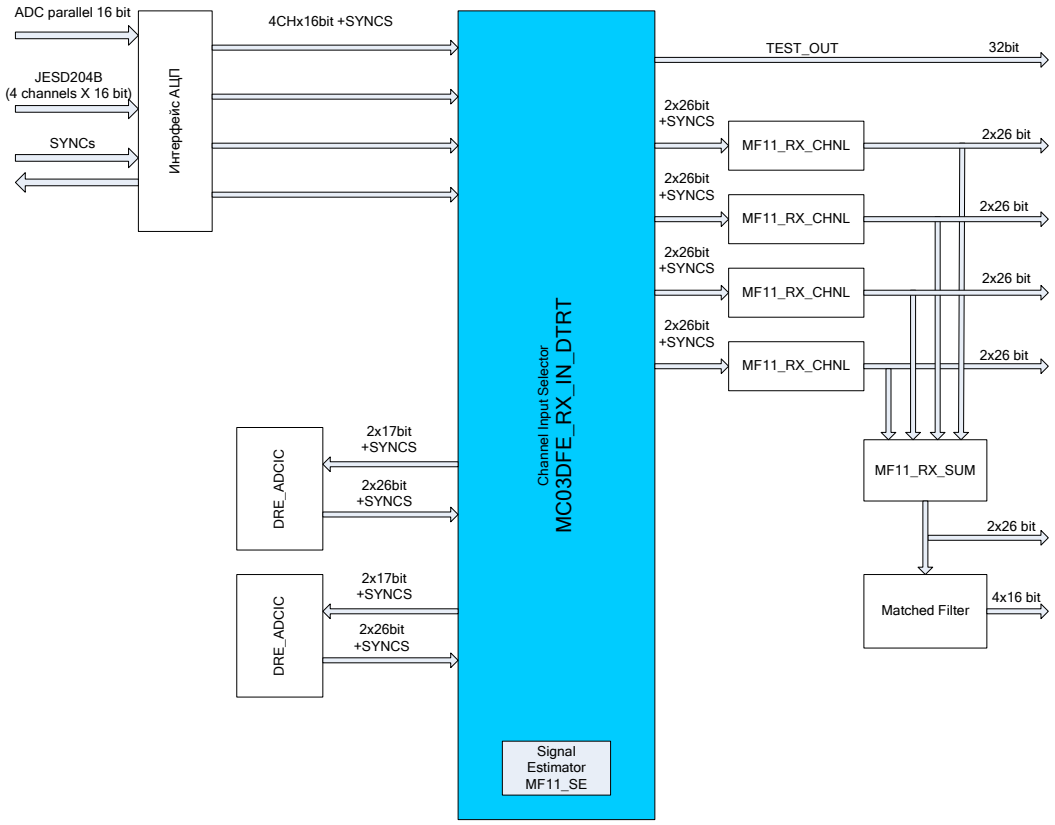


Рисунок 21.6. Функциональная схема RX части DFE_ALT

Интерфейс АЦП предназначен для приема данных от АЦП по параллельной 16-битной шине в домене тактового сигнала АЦП в режимах SDR или DDR с мультиплексированием, приема до четырех потоков данных по интерфейсу JESD204B и формирования потока данных в тактовом домене обработки CLK.

Блоки расширения динамического диапазона АЦП и компенсации небаланса каналов DRE_ADCIC, в зависимости от режима, воспринимают свои входные потоки либо как оцифрованные действительные сигналы от двух разных АЦП с различными коэффициентами усиления, либо как один комплексный сигнал. Каждый блок управляется отдельно, своей группой регистров. В режиме DRE блок формирует один действительный сигнал с большим динамическим диапазоном, в режиме ADCIC – один комплексный сигнал с компенсированным небалансом амплитуд и фаз.

Селектор источников входных сигналов MC03DFE_RX_IN_DTRT предназначен для выбора источника сигнала для блоков обработки приемной части. Блок осуществляет селекцию источника, выполняет оценку параметров сигналов АЦП, осуществляет коррекцию смещения нуля и приведение данных к формату блоков обработки –

квадратурным сигналам разрядностью 2x26 бит. В режиме действительного сигнала Q-компонента выходного сигнала содержит нули. Также, блок имеет возможность генерации ряда тестовых сигналов для подачи на приемные каналы.

Приемные каналы MF11_RX_CHNL осуществляют перенос частоты, децимацию и фильтрацию сигнала. Блок-схема приемного канала представлена на Рисунок 21.7.



Рисунок 21.7. Функциональная схема канала обработки приемной части DFE_ALT

Каждый приемный канал осуществляет последовательную обработку на следующих устройствах:

- цифровой квадратурный гетеродин,
- блок CIC децимации, состоящий из двух каскадов,
- блок полуполосных FIR фильтров-дециматоров, состоящий из трех каскадов,
- два блока FIR фильтров-дециматоров с программируемыми коэффициентами,
- комплексный умножитель для изменения амплитуды и поворота фазы сигнала.

Сумматор каналов MC03DFE_RX_SUM выполняет весовое суммирование выходных сигналов от 1 до 4 приемных каналов, работающих в синхронном режиме. Вес отсчетов каждого канала в сумме задается комплексными умножителями на выходах каналов.

Согласованный фильтр (Matched Filter) MF11_MTCFLT128 осуществляет свертку комплексного сигнала с импульсной характеристикой длиной до 128 комплексных восьмибитных отсчетов. На выход фильтра подаются 4 16-битных числа, представляющие собой скалярные произведения для каждой комбинации реальных и мнимых частей сигнала и импульсной характеристики.

21.1.4.2.2 Передающая часть

Функциональная схема TX части представлена на Рисунок 21.8.

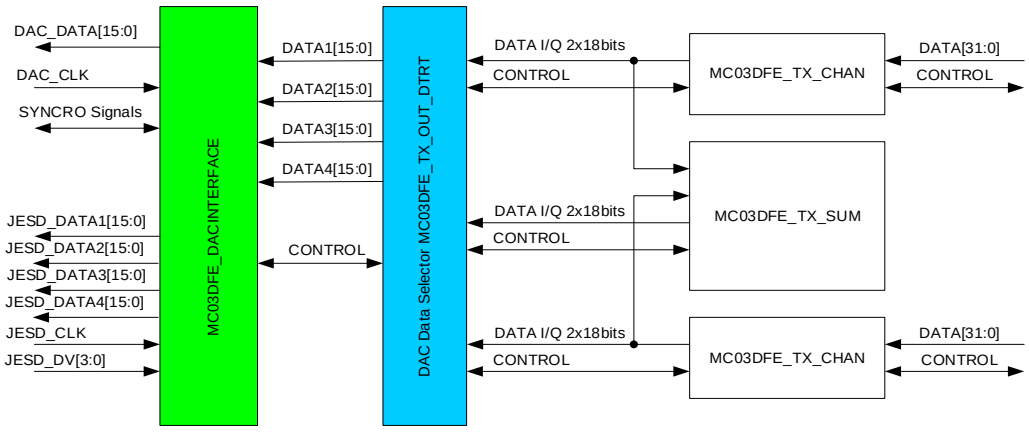


Рисунок 21.8. Функциональная схема TX части DFE_ALT

Интерфейс ЦАП MC03DFE_DACINTERFACE предназначен для передачи во внешний ЦАП до 4 потоков отсчетов сигналов по параллельной 16-битной шине в режимах SDR или DDR с мультиплексированием, или до четырех потоков данных по интерфейсу JESD.

Блок может формировать до двух независимых каналов обработки квадратурного сигнала и поддерживает одновременно максимум два потребителя сигнала (ИМС ЦАП). Обработка заключается в приведении отсчетов к разрядности и кодировке (прямо-смещенной или двоично-дополнительной), поддерживаемым внешними ИМС ЦАП, а также переводе в домен тактовой частоты ЦАП.

Селектор источников входных сигналов для ЦАП MC03DFE_TX_OUT_DTRT. Блок предназначен для выбора источника сигнала для каждого ЦАП. Источником может быть один из TX каналов или весовой сумматор.

Передающие каналы MC03DFE_TX_CHAN осуществляет модуляцию, канальную фильтрацию, интерполяцию и перенос частоты аналитического сигнала. Блок-схема одного канала приемного тракта представлена на Рисунок 21.9.

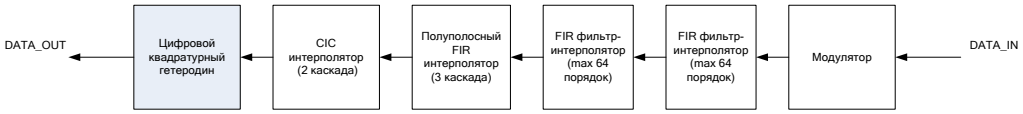


Рисунок 21.9. Функциональная схема канала обработки передающей части DFE_ALT

Каждый передающий канал осуществляет последовательную обработку на следующих устройствах:

- модулятор с функцией расширения спектра;
- два каскада FIR фильтров-интерполяторов с программируемыми коэффициентами;
- блок полуполосных FIR фильтров-интерполяторов, состоящих из трех каскадов;
- блок CIC интерполяции, состоящий из двух каскадов;
- цифровой квадратурный гетеродин.

Сумматор каналов MC03DFE_TX_SUM выполняет весовое суммирование двух передающих каналов TX_CHAN, работающих в синхронном режиме.

Внутренняя разрядность данных межблочного обмена в передатчике – 2x18 бит.

21.1.4.3 Ограничения

При работе на скоростях потока, близких к максимальным, необходимо учитывать ограничение в кросс-клокинговом FIFO, при котором соотношение частот обработки к частоте интерфейса менее 1,25 приводит к ошибкам. Уменьшение соотношения может быть реализовано за счет увеличения глубины FIFO с 8 до 16 отсчетов.

21.2 Регистры

21.2.1 Регистры блока DMA

Перечень регистров DMA приведен в Таблица 21.1.

Таблица 21.1. Перечень программно-доступных регистров

Условное Обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
ir0	Регистр индекса 0-го канала DMA (64 бита)	0x0	W/R	0x00
cp0	Регистр указателя цепочки 0-го канала DMA (64 бита)	0x0	W/R	0x08
csr0	Регистр управления и состояния 0-го канала DMA	0x0	W/R	0x10
run0	Псевдорегистр управления состоянием бита RUN регистра CSR 0-го канала DMA	0x0	W/R	0x14
ir1	Регистр индекса 1-го канала DMA (64 бита)	0x0	W/R	0x20
cp1	Регистр указателя цепочки 1-го канала DMA (64 бита)	0x0	W/R	0x28
csr1	Регистр управления и состояния 1-го канала DMA	0x0	W/R	0x30
run1	Псевдорегистр управления состоянием бита RUN регистра CSR 1-го канала DMA	0x0	W/R	0x34
ir2	Регистр индекса 2-го канала DMA (64 бита)	0x0	W/R	0x40
cp2	Регистр указателя цепочки 2-го канала DMA (64 бита)	0x0	W/R	0x48
csr2	Регистр управления и состояния 2-го канала DMA	0x0	W/R	0x50
run2	Псевдорегистр управления состоянием бита RUN регистра CSR 2-го канала DMA	0x0	W/R	0x54
ir3	Регистр индекса 3-го канала DMA (64 бита)	0x0	W/R	0x60
cp3	Регистр указателя цепочки 3-го канала DMA (64 бита)	0x0	W/R	0x68
csr3	Регистр управления и состояния 3-го канала DMA	0x0	W/R	0x70
run3	Псевдорегистр управления состоянием бита RUN регистра CSR 3-го канала DMA	0x0	W/R	0x74

21.2.1.1 Регистр индекса IR

Формат регистра IR приведен в Таблица 21.2.

Таблица 21.2. Формат регистра IR

Номер бита	Условное обозначение	Назначение
63:0	IR	IR содержат физический адрес внутренней или внешней памяти. После передачи каждого слова данных к индексу IR прибавляется смещение на одно 64-х разрядное слово (32-х разрядное в nels_dma_cmd).

21.2.1.2 Регистр указателя цепочки CP

Формат регистра CP приведен в Таблица 21.3.

Таблица 21.3. Формат регистра CP

Номер бита	Условное Обозначение	Назначение
0	INIT	Бит запуска процедуры самоинициализации
63:1	CP	Начальный адреса блока параметров DMA передачи для самоинициализации

21.2.1.3 Регистр управления и состояния CSR

Формат регистра CSR приведен в Таблица 21.4.

Таблица 21.4. Формат регистра CSR

Номер бита	Условное Обозначение	Назначение
0	RUN	Состояние работы канала DMA: 0 – состояние останова; 1 – состояние обмена данными. Устанавливается в 1 при записи 1 в этот разряд. Устанавливается в 0 при завершении передачи блока данных. Состояние этого бита определяется в процессе выполнения процедуры самоинициализации
1	-	Резерв
5:2	WN	Число слов данных (пачка), которое передается за одно предоставление прямого доступа: 0 – 1 слово, F – 16 слов. Посредством этого параметра можно плавно изменять приоритет каналов DMA относительно других устройств и относительно друг Друга
11:6	-	Резерв
12	CHEN	Разрешение выполнения очередной процедуры самоинициализации: 0 – выполнение очередной процедуры самоинициализации запрещено; 1 – выполнение очередной процедуры самоинициализации разрешено. Используется только при обмене цепочкой блоков данных. Состояние этого бита определяется в процессе выполнения процедуры самоинициализации
13	IM	Маска разрешение установки признака END: 0 – установки признака запрещено; 1 – установки признака разрешено. Используется только при обмене цепочкой блоков данных. Состояние этого бита определяется в процессе выполнения процедуры самоинициализации

Номер бита	Условное Обозначение	Назначение
14	END	Признак завершения передачи блока данных. Устанавливается в 1 при завершении передачи блока данных (при IM=1). Устанавливается в 0 при чтении содержимого этого регистра.
15	DONE	Признак завершения передачи блока данных. Устанавливается в 1 при завершении передачи блока данных при CHEN=0 (CHEN=1 может быть только при использовании процедуры
31:16	WCX	Число 64-разрядных (32-разрядных для Nels_dma_cmd) слов данных, которые должен передать канал DMA (блок данных). Количество передаваемых слов: WCX + 1 Содержимое этого поля уменьшается на 1 после передачи каналом DMA очередного слова данных.

21.2.1.4 Псевдорегистр управления состоянием бита RUN регистра CSR

Состоянием разряда 0 регистра CSR можно управлять, используя адрес псевдорегистра RUN. При этом остальные разряды этого регистра CSR не изменяются. Эта процедура может быть использована для временной приостановки канала DMA. При чтении по адресу псевдорегистра RUN считывается содержимое регистра CSR без сброса битов END и DONE.

Бит RUN может быть использован для остановки работы канала DMA портов. Для этого в любой момент времени в него необходимо записать 0. Эта процедура возможна, если длина массива данных, указанного в канале DMA порта, равна длине массива данных, который порт передаст. Для продолжения работы в бит RUN необходимо записать 1.

Формат псевдорегистра RUN приведен в Таблица 21.5.

Таблица 21.5. Формат псевдорегистра RUN

Номер бита	Условное обозначение	Назначение
0	RUN	Дублирует бит Run регистра CSR
31:1	-	Резерв

21.2.2 Регистры блока DFE_ALT

Данный раздел является справочным. Он содержит перечень регистров блока DFE_ALT и краткое описание к каждому из них, включающее назначение и конфигурацию полей. Для ознакомления с устройством блока и принципом его работы рекомендуется перейти к Разделу «Функционирование» и его подразделу «DFE_ALT».

Регистры, находящиеся в диапазоне смещений 0xA000-0xAFFF и 0x12000-0x12FFF адресуются в страничном или групповом режиме. Выбор режима и страницы осуществляется через регистр RX_CTRL для адресов 0xA000-0xAFFF и через регистр TX_CTRL для адресов 0x12000-0x12FFF.

Номер страницы, который задается в поле CH_CFG_SEL регистра RX_CTRL, соответствует номеру приемного канала.

Номер передающего канала и соответствующая ему страница задаются в поле CH_CFG_SEL регистра TX_CTRL.

Групповой режим выбирается установкой бита CFG_WR_ALL регистров RX_CTRL и TX_CTRL. При его включении каналы конфигурируются синхронно одинаковыми данными, выбор страницы игнорируется, а читаемые данные не валидны.

Таблица 21.6. Перечень программно-доступных регистров

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
DEVID	Регистр идентификатора устройства	0x00000103	R	0x0000
CLK_SYNC_CFG	Управление синхросигналами и тактовыми сигналами	0x00000000	RW	0x0004
SYNC_CTRL	Управление синхросигналами	0x00000000	RW	0x0008
DMARX12_CTRL	Управление 1 и 2 каналом RX DMA	0x00000000	RW	0x000C
DMARX34_CTRL	Управление 3 и 4 каналом RX DMA	0x00000000	RW	0x0010
DMARX56_CTRL	Управление 5 и 6 каналом RX DMA	0x00000000	RW	0x0014
DMATX12_CTRL	Управление 1 и 2 каналом TX DMA	0x00000303	RW	0x0018
IRQ	Регистр запросов прерываний	0x00000000	RW	0x0080
IRQ_MASK	Регистр маскирования прерываний	0x00000000	RW	0x0084
IRQ_RESET	Регистр сброса прерываний	0x00000000	RW	0x0088
IRQ_SENSE	Регистр управления чувствительностью прерываний	0x0000FF9F	RW	0x008C
DMA_INT	Регистр запросов прерываний от DMA	0x00000300	RW	0x0090
DMA_INT_MASK	Регистр маскирования прерываний от DMA	0x00000000	RW	0x0094
DMA_INT_RESET	Регистр сброса прерываний от DMA	0x00000000	RW	0x0098
DMA_INT_SENSE	Регистр управления чувствительностью прерываний от DMA	0x003FFFFF	RW	0x009C
LP_INT	Регистр запросов низкоприоритетных прерываний	0x00000000	RW	0x00A0
LP_INT_MASK	Регистр маски низкоприоритетных прерываний	0x00000000	RW	0x00A4
LP_INT_RESET	Регистр сброса низкоприоритетных прерываний	0x00000000	RW	0x00A8
LP_INT_SENSE	Регистр управления чувствительностью низкоприоритетных прерываний	0x0000F3FF	RW	0x00AC
PROBE_CTRL	Регистр управления PROBE	0x000006C0	RW	0x0100
PROBE_DATA	Регистр данных PROBE	0x00000000	RO	0x0104
SCHED0_SYNCCFG_CTRL	Регистр состояния и управления планировщиком конфигурации 0	0x00000000	RW	0x0180
SCHED0_CNT	Счетчик отсчетов	0x1BAD0BAD	RW	0x0184
SCHED0_TASK	Запись: целевой счетчик новой задачи Чтение: целевой счетчик текущей задачи	0x1BAD0BAD	RW	0x0188
SCHED1_SYNCCFG_CTRL	Регистр состояния и управления планировщиком конфигурации 1	0x00000000	RW	0x0280

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
SCHED1_CNT	Счетчик отсчетов	0x1BAD0BAD	RW	0x0284
SCHED1_TASK	Запись: целевой счетчик новой задачи Чтение: целевой счетчик текущей задачи	0x1BAD0BAD	RW	0x0288
RX_CTRL	Управление страничной адресацией регистров управления приемных каналов в диапазоне смещений 0xA000-0xAFFF	0x00000000	RW	0x8000
ADC_CTRL	Режим работы интерфейса с АЦП	0x00000000	RW, RO	0x8080
DATA_FMT_AUX	Режим форматирования данных в каналах	0x000000F0	RW	0x8084
RX_IN_DTRT_ICH_CTRL	Регистр селекции источников входных сигналов	0x00000000	RW	0x8100
RX_IN_DTRT_OCH12_CTRL	Регистр селекции источников сигнала для выходов 1 и 2 каналов	0x00000000	RW	0x8104
RX_IN_DTRT_OCH34_CTRL	Регистр селекции источников сигнала для выходов 3 и 4 каналов	0x00000000	RW	0x8108
RX_IN_DTRT_DREADCIC_CTRL	Регистр селекции источников сигнала для блоков DREADCIC	0x00000000	RW	0x810C
RX_IN_DTRT_TST1	Регистр тестового воздействия	0x01FFFFFF	RW	0x8110
RX_IN_DTRT_TST2	Регистр тестового воздействия	0x02000001	RW	0x8114
RX_IN_DTRT_TST_CTRL	Регистр селекции источника сигнала для тест-сигналов и блока оценки сигналов	0x0000800F	RW,	0x8118
RX_IN_DTRT_OVF_STATUS	Регистр состояния перегрузок	0x00000000	RO	0x811C
RX_IN_DTRT_INCH_OFFSET0	Установка смещения нуля в канале 0	0x00000000	RW	0x8120
RX_IN_DTRT_INCH_OFFSET1	Установка смещения нуля в канале 1	0x00000000	RW	0x8124
RX_IN_DTRT_INCH_OFFSET2	Установка смещения нуля в канале 2	0x00000000	RW	0x8128
RX_IN_DTRT_INCH_OFFSET3	Установка смещения нуля в канале 3	0x00000000	RW	0x812C
RX_IN_DTRT_INCH_OVFTRS0	Порог срабатывания сигнала перегрузки в канале 0	0x00000000	RW	0x8130
RX_IN_DTRT_INCH_OVFTRS1	Порог срабатывания сигнала перегрузки в канале 1	0x00000000	RW	0x8134
RX_IN_DTRT_INCH_OVFTRS2	Порог срабатывания сигнала перегрузки в канале 2	0x00000000	RW	0x8138
RX_IN_DTRT_INCH_OVFTRS3	Порог срабатывания сигнала перегрузки в канале 3	0x00000000	RW	0x813C
RX_IN_DTRT_SE_POWRMS	Оценка мощности сигнала	0x00000000	RO	0x8140
RX_IN_DTRT_SE_MIN	Оценка минимального значения сигнала	0x01FFFFFF	RO	0x8144
RX_IN_DTRT_SE_MAX	Оценка максимального значения сигнала	0x02000000	RO	0x8148
RX_IN_DTRT_SE_MEAN	Оценка среднего значения сигнала	0x00000000	RO	0x814C
RX_SUM_CTRL	Управление сумматором	0x00000300	RW, RO	0x8180
DRE_ADCIC0_CONTROL_STATUS	Регистр управления и статуса DRE_ADCIC0	0x0000008F	RW, WO, RO	0x8200
DRE_ADCIC0_GAIN_ERR	Выходной сигнал ошибки оценки усиления DRE_ADCIC0	0x00000000	RO	0x8204
DRE_ADCIC0_OFFSET_ERR	Выходной сигнал ошибки оценки нуля DRE_ADCIC0	0x00000000	RO	0x8208
DRE_ADCIC0_K1	Коэффициент K1 для DRE_ADCIC0	0x00000000	RW	0x820C
DRE_ADCIC0_K2	Коэффициент K2 для DRE_ADCIC0	0x00000000	RW	0x8210

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
DRE_ADCIC0_K3	Коэффициент K3 для DRE_ADCIC0	0x00000000	RW	0x8214
DRE_ADCIC0_DRE_GAIN_INT	Текущее значение усиления в режиме авто-DRE в DRE_ADCIC0. Целая часть.	0x00000000	RO	0x8218
DRE_ADCIC0_DRE_GAIN_FRC	Текущее значение усиления в режиме авто-DRE в DRE_ADCIC0. Дробная часть.	0x00000000	RO	0x821C
DRE_ADCIC1_CONTROL_STATUS	Регистр управления и статуса DRE_ADCIC1	0x0000008F	RW, WO, RO	0x8280
DRE_ADCIC1_GAIN_ERR	Выходной сигнал ошибки оценки усиления DRE_ADCIC1	0x00000000	RO	0x8284
DRE_ADCIC1_OFFSET_ERR	Выходной сигнал ошибки оценки нуля DRE_ADCIC1	0x00000000	RO	0x8288
DRE_ADCIC1_K1	Коэффициент K1 для DRE_ADCIC1	0x00000000	RW	0x828C
DRE_ADCIC1_K2	Коэффициент K2 для DRE_ADCIC1	0x00000000	RW	0x8290
DRE_ADCIC1_K3	Коэффициент K3 для DRE_ADCIC1	0x00000000	RW	0x8294
DRE_ADCIC1_DRE_GAIN_INT	Текущее значение усиления в режиме авто-DRE в DRE_ADCIC1. Целая часть.	0x00000000	RO	0x8298
DRE_ADCIC1_DRE_GAIN_FRC	Текущее значение усиления в режиме авто-DRE в DRE_ADCIC1. Дробная часть.	0x00000000	RO	0x829C
MTCFLT_CORE0_TAP0	Коэффициенты согласованного фильтра для вычислительного ядра 0	0x00000000	RW	0x8400
MTCFLT_CORE0_TAP1		0x00000000	RW	0x8404
……		……	……	……
MTCFLT_CORE0_TAP31		0x00000000	RW	0x847C
MTCFLT_CORE1_TAP0	Коэффициенты согласованного фильтра для вычислительного ядра 1	0x00000000	RW	0x8480
MTCFLT_CORE1_TAP1		0x00000000	RW	0x8484
……		……	……	……
MTCFLT_CORE1_TAP31		0x00000000	RW	0x84FC
MTCFLT_CORE2_TAP0	Коэффициенты согласованного фильтра для вычислительного ядра 2	0x00000000	RW	0x8500
MTCFLT_CORE2_TAP1		0x00000000	RW	0x8504
……		……	……	……
MTCFLT_CORE2_TAP31		0x00000000	RW	0x857C
MTCFLT_CORE3_TAP0	Коэффициенты согласованного фильтра для вычислительного ядра 3	0x00000000	RW	0x8580
MTCFLT_CORE3_TAP1		0x00000000	RW	0x8584
……		……	……	……
MTCFLT_CORE3_TAP31		0x00000000	RW	0x85FC
MTCFLT_CFG	Регистр конфигурации согласованного фильтра MTCFLT128	0x00000000	RW	0x8600
RX_CHNL_CTRL	Общий регистр управления канала	0x00000000	WC, RO,RW	0xA000
RX_CHNL_PPS_POS	Позиция строба PPS относительно начала входного потока.	0xFFFFFFFF		0xA004
RX_NCO_FRQ	Частота гетеродина	0x00000000	RW	0xA080
RX_NCO_SWRATE	Приращение частоты гетеродина	0x00000000	RW	0xA084
RX_NCO_SWTIME	Период свипа	0xFFFFFFFF	RW	0xA088

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
RX_NCO_PHASE	Смещение фазы гетеродина	0x00000000	RW	0xA08C
RX_NCO_MODE	Режим работы гетеродина	0x00000000	RW	0xA090
DCIC2_CFG	Управление DCIC2	0x00000000	RW	0xA100
DCICN_CFG	Управление DCICN	0x00000040	RW	0xA104
DCIC_BWGAUGE	Индикаторы уровня сигнала DCIC2 и DCICN	0x00000000	WC	0xA108
DHB_MODE	Режим работы полуполосных фильтров	0x00003000	RW, WO	0xA180
DHB_BWGAUGE	Индикатор уровня сигнала DHB	0x00000000	WC	0xA184
RX_FGAIN_COFF	Комплексный коэффициент усиления	0x20000000	RW	0xA200
RX_FGAIN_CTRL	Регистр управления устройством масштабирования и поворота фазы	0x00008300	WC, RW, RO	0xA204
DFIR0_P0_CF0	Коэффициенты профиля 0 фильтра DFIR0 с нулевого по 31	0x00000000	RW	0xA400
DFIR0_P0_CF1		0x00000000	RW	0xA404
…….		…….	…….	…….
DFIR0_P0_CF31		0x00000000	RW	0xA47C
DFIR0_P1_CF0	Коэффициенты профиля 1 фильтра DFIR0 с нулевого по 31	0x00000000	RW	0xA480
DFIR0_P1_CF1		0x00000000	RW	0xA484
….		….	….	….
DFIR0_P1_CF31		0x00000000	RW	0xA4FC
DFIR0_PROFILE0	Регистр профиля 0 DFIR0	0x00000000	RW	0xA500
DFIR0_PROFILE1	Регистр профиля 1 DFIR0	0x00000000	RW	0xA504
DFIR0_CFG	Регистр конфигурации DFIR0	0x00000000	RW	0xA508
DFIR0_BWGAUGE	Оценка уровня сигнала DFIR0	0x00000000	WC	0xA50C
DFIR1_P0_CF0	Коэффициенты профиля 0 фильтра DFIR1 с нулевого по 31	0x00000000	RW	0xA800
DFIR1_P0_CF1		0x00000000	RW	0xA804
…….		…….	…….	…….
DFIR1_P0_CF31		0x00000000	RW	0xA87C
DFIR1_P1_CF0	Коэффициенты профиля 1 фильтра DFIR1 с нулевого по 31	0x00000000	RW	0xA880
DFIR1_P1_CF1		0x00000000	RW	0xA884
….		….	….	….
DFIR1_P1_CF31		0x00000000	RW	0xA8FC
DFIR1_PROFILE0	Регистр профиля 0 DFIR1	0x00000000	RW	0xA900
DFIR1_PROFILE1	Регистр профиля 1 DFIR1	0x00000000	RW	0xA904
DFIR1_CFG	Регистр конфигурации DFIR1	0x00000000	RW	0xA908
DFIR1_BWGAUGE	Оценка уровня сигнала DFIR1	0x00000000	WC	0xA90C
TX_CTRL	Управление страничной адресацией регистров управления передающих каналов в диапазоне смещений 0x12000-0x12FFF	0x00000000	RW	0x10000
DAC_CTRL	Режим работы интерфейса с ЦАП	0x00000040	RW, RO	0x10080
TX_OUT_DTRT_ROUTE_CTRL	Регистр селекции выходных сигналов	0x00000000	RW	0x10100
TX_OUT_DTRT_OVF_STATUS	Регистр индикации перегрузок	0x00000000	RO	0x10104
TX_OUT_DTRT_TST1	Регистр тестового воздействия 1	0x00007FFF	RW	0x10108
TX_OUT_DTRT_TST2	Регистр тестового воздействия 2	0x00008001	RW	0x1010C
TX_OUT_DTRT_CH1_OFFSET	Установка смещения нуля в 1 канале	0x00000000	RW	0x10110
TX_OUT_DTRT_CH2_OFFSET	Установка смещения нуля в 2 канале	0x00000000	RW	0x10114
TX_OUT_DTRT_CH3_OFFSET	Установка смещения нуля в 3 канале	0x00000000	RW	0x10118
TX_OUT_DTRT_CH4_OFFSET	Установка смещения нуля в 4 канале	0x00000000	RW	0x1011C

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
TX_OUT_DTRT_TST_CTRL	Выбор источников для записи через PROBE и управление блоком оценки SE	0x0000800F	RW	0x10120
TX_OUT_DTRT_SE_POWRMS	Оценка мощности сигнала	0x00000000	RO	0x10124
TX_OUT_DTRT_SE_MIN	Оценка минимального значения сигнала	0x00007FFF	RO	0x10128
TX_OUT_DTRT_SE_MAX	Оценка максимального значения сигнала	0x00008000	RO	0x1012C
TX_OUT_DTRT_SE_MEAN	Оценка среднего значения сигнала	0x00000000	RO	0x10130
TX_SUM_CTRL	Управление сумматором	0x00000003	RW, RO	0x10200
TX_CHNL_CTRL	Общий регистр управления передающим каналом	0x00000000	WC, RW, RO	0x12000
TX_CHNL_TASK_STATE	Регистр управления планировщиком старта передачи	0x00000000	RW, WC	0x12004
TX_CHNL_DT1	Установка защитного интервала dT1	0x00000000	RW	0x12008
TX_CHNL_DT2	Установка защитного интервала dT2	0x00000000	RW	0x1200C
TX_CHNL_TASK_TME	Запись: целевой счетчик новой задачи Чтение: целевой счетчик текущей задачи	0x00000000	RW	0x12010
TX_NCO_FRQ	Частота гетеродина	0x00000000	RW	0x12080
TX_NCO_SWRATE	Приращение частоты гетеродина	0x00000000	RW	0x12084
TX_NCO_SWTIME	Период свипа	0xFFFFFFFF	RW	0x12088
TX_NCO_PHASE	Смещение фазы гетеродина	0x00000000	RW	0x1208C
TX_NCO_MODE	Режим работы гетеродина	0x00000000	RW	0x12090
ICIC_CIC2_CFG	Управление ICIC2	0x00000000	RW, WC	0x12100
ICIC_CICN_CFG	Управление ICICN	0x00000000	RW	0x12104
ICIC_BWGAUGE	Индикаторы уровня сигнала ICICN и ICIC2	0x00000000	WC	0x12108
IHB_MODE	Управление полуполосными фильтрами	0x00003000	RW, WO	0x12180
IHB_BWGAUGE	Индикаторы уровня сигнала IHB	0x00000000	WC	0x12184
MOD_MODE	Режим работы модулятора	0x00000000	RW	0x12280
MOD_PN_PROFILE	Регистр выбора профиля модулятора	0x00000000	RW	0x12284
IFIR0_P0_CF0	Коэффициенты профиля 0 фильтра IFIR0 с нулевого по 31	0x00000000	RW	0x12400
IFIR0_P0_CF1		0x00000000	RW	0x12404
…..		…..		…..
IFIR0_P0_CF31		0x00000000	RW	0x1247C
IFIR0_P1_CF0	Коэффициенты профиля 1 фильтра IFIR0 с нулевого по 31	0x00000000	RW	0x12480
IFIR0_P1_CF1		0x00000000	RW	0x12484
…..		…..	…..	…..
IFIR0_P1_CF31		0x00000000	RW	0x124FC
IFIR0_PROFILE0	Регистр профиля 0 IFIR0	0x00000000	RW	0x12500
IFIR0_PROFILE1	Регистр профиля 1 IFIR0	0x00000000	RW	0x12504
IFIR0_CFG	Регистр конфигурации IFIR0	0x00000000	RW	0x12508
IFIR0_BWGAUGE	Оценка уровня сигнала IFIR0	0x00000000	WC	0x1250C
IFIR1_P0_CF0	Коэффициенты профиля 0 фильтра IFIR1 с нулевого по 31	0x00000000	RW	0x12800
IFIR1_P0_CF1		0x00000000	RW	0x12804
…..		…..	…..	…..
IFIR1_P0_CF31		0x00000000	RW	0x1287C
IFIR1_P1_CF0	Коэффициенты профиля 1 фильтра IFIR1 с нулевого по 31	0x00000000	RW	0x12880
IFIR1_P1_CF1		0x00000000	RW	0x12884
…..		…..	…..	…..
IFIR1_P1_CF31		0x00000000	RW	0x128FC
IFIR1_PROFILE0	Регистр профиля 0 IFIR1	0x00000000	RW	0x12900
IFIR1_PROFILE1	Регистр профиля 1 IFIR1	0x00000000	RW	0x12904
IFIR1_CFG	Регистр конфигурации IFIR1	0x00000000	RW	0x12908
IFIR1_BWGAUGE	Оценка уровня сигнала IFIR1	0x00000000	WC	0x1290C
MOD_MEM0		0x00000000	RW	0x12C00

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
MOD_MEM1	Регистры чтения и записи памяти модулятора (256 слов по 16 бит)	0x00000000	RW	0x12C04
MOD_MEM2		0x00000000	RW	0x12C08
…..		…..	…..	…..
MOD_MEM255		0x00000000	RW	0x12FFC

21.2.2.1 Регистр DEVID

Назначение: получение информации об устройстве и его версии.

Таблица 21.7. Формат регистра DEVID

Номер бита	Условное обозначение	Назначение
31:0	DEVID	Идентификатор устройства

21.2.2.2 Регистр CLK_SYNC_CFG

Назначение: управление сигналами синхронизации АЦП и ЦАП, точная подстройка фазы тактовых сигналов АЦП и ЦАП.

Таблица 21.8. Формат регистра CLK_SYNC_CFG

Номер бита	Условное обозначение	Назначение
31:16	–	Резерв
15	FB_CLK_ON	Включение тактового сигнала на тестовом выходе. 1 - включен
14:13	–	Резерв
12	DAC_CLKINV	Управление инверсией тактового сигнала DAC_CLK. Воздействует одновременно на все инверторы блоков подстройки фазы тактового сигнала. 1 - входной тактовый сигнал инвертируется.
11:8	DAC_CLKDLY	Управление задержкой тактового сигнала DAC_CLK. Воздействует одновременно на все инверторы блоков подстройки фазы тактового сигнала. 1 шаг подстройки соответствует ~100pS
7	RX_FB_CLK_ON	Включение тактового сигнала на тестовом выходе. 1 - включен
6:5	–	Резерв
4	ADC_CLKINV	Управление инверсией тактового сигнала ADC_CLK. Воздействует одновременно на все инверторы блоков подстройки фазы тактового сигнала. 1 - входной тактовый сигнал инвертируется.
3:0	ADC_CLKDLY	Управление задержкой тактового сигнала ADC_CLK. Воздействует одновременно на все инверторы блоков подстройки фазы тактового сигнала. 1 шаг подстройки соответствует ~100pS

21.2.2.3 Регистр SYNC_CTRL

Назначение: управление сигналами синхронизации старта и применения конфигурации.

Таблица 21.9. Формат регистра SYNC_CTRL

Номер бита	Условное обозначение	Назначение
31:16	–	Резерв
15	PPS_SEL	Выбор источника шкалы времени (сигнала PPS) 0 - старт синхронен с PPS[0] 1 - старт синхронен с PPS[1]
14	–	Резерв
13	RX_START_PPSSYNC	Синхронизация старта RX тракта с импульсом PPS 0 - синхронизации нет. 1 - старт синхронен с выбранным PPS
12	TX_START_PPSSYNC	Синхронизация старта TX тракта с импульсом PPS 0 - синхронизации нет. 1 - старт синхронен с выбранным PPS
11:8	SYNCCFGOUT_DLY	Длительность строба SYNC_CFG на тестовом выходе в тактах, уменьшенная на 1.
7:3	–	Резерв
2	RX_SYNCSTARTOUT	Установка значения сигнала SYNC_START_OUT RX тракта.
1	TX_SYNCSTARTOUT	Установка значения сигнала SYNC_START_OUT TX тракта.
0	RX_SYNCCFGOUT	Установка значения сигнала SYNC_CFG_OUT RX тракта.

21.2.2.4 Регистры DMARX12_CTRL, DMARX34_CTRL, DMARX56_CTRL

Назначение: группа регистров для управления каналами DMA приемного тракта.

Каждый регистр управляет парой каналов DMA приемного тракта. Формат регистра приведен в таблице на примере регистра DMARX12_ CTRL.

Таблица 21.10. Формат регистра DMARX12_CTRL

Номер бита	Условное обозначение	Назначение
31:16	–	Резерв
15	DMARX1_ENA	Включение/выключение канала DMA. Активный - единица.
14:12	DMARX1_SRC	Выбор источника данных. 0 - сигнал DMA_ENA соответствующего канала DMA не активен. 1 - шина данных подключена к выходу RXCH1, сигнал DMA_ENA соответствующего канала DMA активен. 2 - шина данных подключена к выходу RXCH2, сигнал DMA_ENA соответствующего канала DMA активен. 3 - шина данных подключена к выходу RXCH3, сигнал DMA_ENA соответствующего канала DMA активен. 4 - шина данных подключена к выходу RXCH4, сигнал DMA_ENA соответствующего канала DMA активен. 5 - шина данных подключена к выходу RX SUM, сигнал DMA_ENA соответствующего канала DMA активен. 6 - шина данных подключена к выходу RX MTFLT, сигнал DMA_ENA соответствующего канала DMA активен. 7 - шина данных подключена к выходу PROBE, сигнал DMA_ENA соответствующего канала DMA активен.
11:10	DMARX1_PACK_MODE	Управление режимом упаковки данных
9	DMARX1_FULL	Флаги состояния «Full» FIFO канала DMA. Активный – единица (RO)

Номер бита	Условное обозначение	Назначение
8	DMARX1_NEMPTY	Флаги состояния «Not Empty» FIFO канала DMA. Активный – единица (RO)
7	DMARX2_ENA	Включение/выключение канала DMA. Активный - единица.
6:4	DMARX2_SRC	Выбор источника данных. Аналогично DMARX1_SRC
3:2	DMARX2_PACK_MODE	Управление режимом упаковки данных
1	DMARX2_FULL	Флаги состояния «Full» FIFO канала DMA. Активный – единица (RO)
0	DMARX2_NEMPTY	Флаги состояния «Not Empty» FIFO канала DMA. Активный – единица (RO)

21.2.2.5 Регистр DMATX12_CTRL

Назначение: управление каналами DMA передающего тракта.

Таблица 21.11. Формат регистра DMATX12_CTRL

Номер бита	Условное обозначение	Назначение
31:16	–	Резерв
15	DMATX1_ENA	Включение/выключение канала DMA. Активный - единица.
14:10	–	Резерв
9	DMATX1_EMPTY	Флаги состояния «Empty» FIFO канала DMA. Активный – единица (RO)
8	DMATX1_NFULL	Флаги состояния «Not Full» FIFO канала DMA. Активный – единица (RO)
7	DMATX2_ENA	Включение/выключение канала DMA. Активный - единица.
6:2	–	Резерв
1	DMATX2_EMPTY	Флаги состояния «Empty» FIFO канала DMA. Активный – единица (RO)
0	DMATX2_NFULL	Флаги состояния «Not Full» FIFO канала DMA. Активный – единица (RO)

21.2.2.6 Регистр IRQ

Назначение: отображение флагов запросов на прерывание от различных устройств.

Таблица 21.12. Формат регистра IRQ

Номер бита	Условное обозначение	Назначение
31:16	–	Резерв
15	DMA_INT	Объединенное прерывание от прерываний DMA
14	RX_INFIFO_ERR	Прерывание от MC03DFE_ADCINTERFACE
13	TX_OUTFIFO_ERR	Прерывание от MC03DFE_DACINTERFACE
12	SUM_SYNC_ERR	Прерывание по рассогласованию синхронизации в RX каналах с блока сумматоров каналов
11	PPS_EVENT	Прерывание по PPS
10	TX1_TASK_END	Прерывание по окончании передачи в канале TX1 (из блока TX_CHNL)
9	TX1_TASK_MISS	Прерывание при пропуске передачи фрейма в канале TX1(из блока TX_CHNL)
8	TX2_TASK_END	Прерывание по окончании передачи в канале TX2 (из блока TX_CHNL)

Номер бита	Условное обозначение	Назначение
7	TX2_TASK_MISS	Прерывание при пропуске передачи фрейма в канале TX2 (из блока TX_CHNL)
6:5	–	Резерв
4	TX_PUSH_DONE	Прерывание по окончании процедуры PUSH от MC03DFE_DACINTERFACE
3	DRE_EST_RDY	Объединенное прерывание EST_RDY от блоков MC03DFE_DRE_ADCIC (OVF_EST_INT[15:14])
2	SE_RDY	Объединенное прерывание от сигналов INT_SE блоков MC03DFE_RX_IN_DTRT и MC03DFE_TX_OUT_DTRT (OVF_EST_INT[13:12])
1	PROBE_RDY	Прерывание RDY_INT от MC03DFE_PROBE
0	OVF_INT	Объединенное прерывание от сигналов переполнения “OVF” OVF_EST_INT[8:0]

21.2.2.7 Регистр DMA_INT

Назначение: отображение флагов запросов на прерывание от устройства DMA.

Таблица 21.13. Формат регистра DMA_INT

Номер бита	Условное обозначение	Назначение
31:22	–	Резерв
21	DMARX1_EOF	Прерывание по пустому FIFO (окончанию выборки сигнала) DMARX интерфейса №1
20	DMARX2_EOF	Прерывание по пустому FIFO (окончанию выборки сигнала) DMARX интерфейса №2
19	DMARX3_EOF	Прерывание по пустому FIFO (окончанию выборки сигнала) DMARX интерфейса №3
18	DMARX4_EOF	Прерывание по пустому FIFO (окончанию выборки сигнала) DMARX интерфейса №4
17	DMARX5_EOF	Прерывание по пустому FIFO (окончанию выборки сигнала) DMARX интерфейса №5
16	DMARX6_EOF	Прерывание по пустому FIFO (окончанию выборки сигнала) DMARX интерфейса №6
15	DMARX1_FULL	Прерывание по полному FIFO DMARX интерфейса №1
14	DMARX2_FULL	Прерывание по полному FIFO DMARX интерфейса №2
13	DMARX3_FULL	Прерывание по полному FIFO DMARX интерфейса №3
12	DMARX4_FULL	Прерывание по полному FIFO DMARX интерфейса №4
11	DMARX5_FULL	Прерывание по полному FIFO DMARX интерфейса №5
10	DMARX6_FULL	Прерывание по полному FIFO DMARX интерфейса №6
9	DMATX1_EMP	Прерывание по пустому FIFO DMATX интерфейса №1
8	DMATX2_EMP	Прерывание по пустому FIFO DMATX интерфейса №2
7	DMARX1_ERR	Прерывание по ошибке DMARX интерфейса №1
6	DMARX2_ERR	Прерывание по ошибке DMARX интерфейса №2
5	DMARX3_ERR	Прерывание по ошибке DMARX интерфейса №3
4	DMARX4_ERR	Прерывание по ошибке DMARX интерфейса №4
3	DMARX5_ERR	Прерывание по ошибке DMARX интерфейса №5
2	DMARX6_ERR	Прерывание по ошибке DMARX интерфейса №6
1	DMATX1_ERR	Прерывание по ошибке DMATX интерфейса №1
0	DMATX2_ERR	Прерывание по ошибке DMATX интерфейса №2

21.2.2.8 Регистр LP_INT

Назначение: отображение флагов запросов на прерывание от устройств детектирования перегрузки и оценки сигналов.

Таблица 21.14. Формат регистра LP_INT

Номер бита	Условное обозначение	Назначение
31:16	–	Резерв
15	DRE1_EST_RDY	Прерывание EST_RDY от MC03DFE_DRE_ADCIC
14	DRE2_EST_RDY	Прерывание EST_RDY от MC03DFE_DRE_ADCIC
13	TX_SE_RDY	Прерывание INT_SE от MC03DFE_TX_OUT_DTRT
12	RX_SE_RDY	Прерывание INT_SE от MC03DFE_RX_IN_DTRT
11:9	–	Резерв
8	IRQ_TX1_OVF	Прерывание по перегрузке в канале TX1
7	IRQ_TX2_OVF	Прерывание по перегрузке в канале TX2
6	MTCFLT_OVF	Прерывание по перегрузке в блоке согласованного фильтра
5	SUM_OVF	Прерывание по перегрузке в канале сумматора
4	ADC_OVF	Прерывание по перегрузке (превышению заданного порога) одного из каналов АЦП из RX_IN_DTRT.
3	RX1_OVF	Прерывание по перегрузке в канале RX1
2	RX2_OVF	Прерывание по перегрузке в канале RX2
1	RX3_OVF	Прерывание по перегрузке в канале RX3
0	RX4_OVF	Прерывание по перегрузке в канале RX4

21.2.2.9 Регистры IRQ_MASK, DMA_INT _MASK, LP_INT _MASK

Назначение: маскирование прерываний от различных событий и устройств.

Позиции битов, соответствующих прерываниям от разных событий, совпадают с позициями в регистрах IRQ для IRQ_MASK, DMA_INT для DMA_INT_MASK и LP_INT для LP_INT_MASK.

Единица в соответствующем бите разрешает прерывание.

21.2.2.10 Регистры IRQ_RESET, DMA_INT_RESET, LP_INT_RESET

Назначение: сброс прерываний от различных событий и устройств.

Позиции битов, соответствующих прерываниям от разных событий, совпадают с позициями в регистрах IRQ для IRQ_RESET, DMA_INT для DMA_INT_RESET и LP_INT для LP_INT_RESET.

Единица, записанная в соответствующий бит, сбрасывает прерывание, возникшее по фронту.

21.2.2.11 Регистры IRQ_SENSE, DMA_INT_SENSE, LP_INT_SENSE

Назначение: установка режима прерываний от различных событий и устройств по фронту/уровню.

Значение битов 0 соответствуют режиму по уровню, 1 – по фронту.

Позиции битов, соответствующих прерываниям от разных событий, совпадают с позициями в регистрах IRQ для IRQ_SENSE, DMA_INT для DMA_INT_SENSE и LP_INT для LP_INT_SENSE.

21.2.2.12 Регистр PROBE_CTRL

Назначение: управление устройством записи выборки сигналов.

Таблица 21.15. Формат регистра PROBE_CTRL

Номер бита	Условное обозначение	Назначение
31:16	–	Резерв
15	START	Запись «1» инициирует старт начала записи (WO)
14	DST	Способ чтения записанных данных. 0 - по скоростной шине с использованием DMA 1 - через регистр PROBE_DATA
13:11	–	Резерв
10:8	LNG_REQ	Размер выборки, заказанной для записи 0 - запись блокирована. Размер выборки нулевой. 1 - 0.5k (512 слов) 2 - 1k 3 - 2k 4..7 - 0 (Reserved)
7	STATE	Состояние 0 - выборка в процессе записи. 1 - запись закончена. (RO)
6	IS_EMPTY	Состояние буфера. 1 - Буфер пуст. Данных для чтения нет 0 - В буфере остались данные. (RO)
5	–	Резерв
4:3	SRC	Источник сигнала 0 - Тестовый выход RX тракта 1 - Тестовый выход TX тракта 2 - Зарезервировано 3 - Зарезервировано
2:0	FULLNESS	Размер выборки, которая записана 0 - менее 0.5k (512 слов) 1 - 0.5k (512 слов) 2 - 1k 3 - 2k 4..7 - Reserved

21.2.2.13 Регистр PROBE_DATA

Назначение: чтение отсчетов записанной выборки.

Первое чтение после окончания записи выборки возвращает первый записанный отсчет. Последующие чтения возвращают отсчеты в том же порядке, в каком они были записаны.

21.2.2.14 Регистр SCHEDx_SYNCCFG_CTRL

Назначение: управление планировщиком применения конфигурации передающего канала с номером x.

Таблица 21.16. Формат регистра SCHEDx_SYNCCFG_CTRL

Номер бита	Условное обозначение	Назначение
31:16	–	Резерв
15	SYNC_CFG_ENA	0 - формирование сигнала синхронизации и продвижение очереди запрещено. 1 - Формирование сигнала синхронизации и продвижение очереди разрешено.

Номер бита	Условное обозначение	Назначение
14	CNT_RST	Сброс счетчика отсчетов (WO)
13	SHED_RST	Сброс очереди задач (WO)
12	CNT_LOCK	Запоминание состояния счетчика отсчетов и текущего назначенного момента старта синхронизации
11:4	–	Резерв
3:0	SYNC_CFG_LIST	При чтении возвращается количество занятых ячеек в очереди. 0 соответствует пустой очереди (RO)

21.2.2.15 Регистр SCHEDx_CNT

Назначение: чтение последнего запомненного значения счетчика отсчетов передающего канала с номером x.

Таблица 21.17. Формат регистра SCHEDx_CNT

Номер бита	Условное обозначение	Назначение
31:16	–	Резерв
15:0	CNT	Счетчик отсчетов

21.2.2.16 Регистр SCHEDx_TASK

Назначение: добавление новой задачи синхронизации применения конфигурации в очередь задач передающего канала с номером x. Чтение значения текущей задачи.

Таблица 21.18. Формат регистра SCHEDx_TASK

Номер бита	Условное обозначение	Назначение
31:0	TASK	Запись: добавление в очередь задач синхронизации конфигурации нового целевого значения счетчика отсчетов. Чтение: значение счетчика отсчетов для текущего назначенного задания синхронизации

21.2.2.17 Регистр RX_CTRL

Назначение: выбор режима записи, номера канала (страницы адресации регистров) при конфигурации приемных каналов.

Таблица 21.19. Формат регистра RX_CTRL

Номер бита	Условное обозначение	Назначение
31:3	–	Резерв
2	CFG_WR_ALL	Управление записью конфигурации одновременно во все каналы. 1 - Запись всех каналов. 0 - Запись канала, определяемого CH_CFG_SEL
1:0	CH_CFG_SEL	Выбор канала для записи/чтения регистров управления при страничной адресации

21.2.2.18 Регистр ADC_CTRL

Назначение: управление режимами АЦП и преобразованием формата сигнала.

Таблица 21.20. Формат регистра ADC_CTRL

Номер бита	Условное обозначение	Назначение
31:16	–	Резерв
15:14	SRC_SEL	Глобальное включение/выключение интерфейса, выбор порта. 0 - интерфейс выключен 1 - 16 бит параллельная шина CMOS порта 2 - 16 бит параллельная шина LVDS порта 4 канала JESD204 интерфейса
13:11	PBUS_MODE	0 - Simple Parallel SDR 1 - Simple Parallel DDR мультиплексирование каналов 2 - Simple Parallel DDR мультиплексирование бит 3 - Reserved (Simple Parallel SDR) 4 - JESD207 SDR (AD9361 CMOS compatible) 5 - JESD207 DDR (AD9361 CMOS compatible) 6 - AD9361 DDR LVDS compatible 7 - Reserved (Simple Parallel SDR)
10	MODE_OPT1	Назначение поля зависит от установок SRC_SEL и PBUS_MODE При SRC_SEL=1,2 и PBUS_MODE=0, 1 - разрядность шины АЦП: 0 - 16 бит 1 - 8 бит При PBUS_MODE=2 - порядок следования битов данных: 0 - первыми следуют младшие (четные) биты. 1 - первыми следуют старшие (нечетные) биты. При SRC_SEL=1,2 и PBUS_MODE =4, 5, 6 - количество каналов: 0 - 1 канал 1 - 2 канала
9:8	MODE_OPT2	Разрядность шины данных АЦП для PBUS_MODE=6. 0 - 8 бит 1 - 7 бит 2 - 6 бит 3 - 5 бит
7	CLK_ERR	Признак ошибки перехода между тактовыми доменами активного фронта (RO)
6	FRAME_ERR	Признак ошибки фреймовой синхронизации в режиме JESD207 (RO)
5:3	–	Резерв
2	GRAYDECODE_ON	Включение/выключение декодера Грея 0 - off 1 - on
1	DSCRMB_ON	Включение/выключение дескремблера 0 - off 1 - on
0	OB2TC_ON	Включение/выключение преобразования прямосмещенного кода в двоично -дополнительный 0 - off 1 - on

21.2.2.19 Регистр DATA_FMT_AUX

Назначение: управление дескремблером и маскированием данных на выходе преобразователя форматов.

Таблица 21.21. Формат регистра DATA_FMT_AUX

Номер бита	Условное обозначение	Назначение
31:8	–	Резерв

Номер бита	Условное обозначение	Назначение
7:4	OUT_DATA_BW	Разрядность данных минус 1.
3:0	DSCRMB_BPOS	Позиция бита, по которому выполняется дескремблирование

21.2.2.20 Регистр RX_IN_DTRT_ICH_CTRL

Назначение: управление подключением входных сигналов коммутатора приемного тракта.

Таблица 21.22. Формат регистра RX_IN_DTRT_ICH_CTRL

Номер бита	Условное обозначение	Назначение
31:8	–	Резерв
7:6	ICH4_SRC	Аналогично ICH1_SRC
5:4	ICH3_SRC	Аналогично ICH1_SRC
3:2	ICH2_SRC	Аналогично ICH1_SRC
1:0	ICH1_SRC	0 - Соответствующий каналу сигнал с блока ADC_INTERFACE 1 - Тестовое воздействие TEST_PATTERN_I 2 - Тестовое воздействие TEST_PATTERN_Q 3 - Ноль.

21.2.2.21 Регистры RX_IN_DTRT_OCHx_CTRL

Назначение: управление подключением выходных сигналов в коммутаторе приемного тракта. Два регистра RX_IN_DTRT_OCH12_CTRL и RX_IN_DTRT_OCH34_CTRL управляют каждый своей парой выходов.

Таблица 21.23. Формат регистра RX_IN_DTRT_OCH12_CTRL

Номер бита	Условное обозначение	Назначение
31:15	–	Резерв
14	OCH2_CPLX	0 - Выходной сигнал действительный, Q компонента принудительно обнулена. 1 - Выходной сигнал квадратурный, Q компонента не нулевая.
13:12	OCH2_SEL	См. блок-схему коммутатора DTRT в разделе его описания
11:10	OCH2_I_SEL	См. блок-схему коммутатора DTRT в разделе его описания
9:8	OCH2_Q_SEL	См. блок-схему коммутатора DTRT в разделе его описания
7	–	Резерв
6	OCH1_CPLX	0 - Выходной сигнал действительный, Q компонента принудительно обнулена. 1 - Выходной сигнал квадратурный, Q компонента не нулевая.
5:4	OCH1_SEL	См. блок-схему коммутатора DTRT в разделе его описания
3:2	OCH1_I_SEL	См. блок-схему коммутатора DTRT в разделе его описания
1:0	OCH1_Q_SEL	См. блок-схему коммутатора DTRT в разделе его описания

21.2.2.22 Регистр RX_IN_DTRT_DREADCIC_CTRL

Назначение: управление подключением устройств DREADCIC

Таблица 21.24. Формат регистра RX_IN_DTRT_DREADCIC_CTRL

Номер бита	Условное обозначение	Назначение
31:10	–	Резерв

Номер бита	Условное обозначение	Назначение
9	DREADCIC2_ENA	Разрешение работы DREADCIC2
8	DREADCIC1_ENA	Разрешение работы DREADCIC1
7:6	DREADCIC2_CH2_SEL	Аналогично DREADCIC1_CH1_SRC
5:4	DREADCIC2_CH1_SEL	Аналогично DREADCIC1_CH1_SRC
3:2	DREADCIC1_CH2_SEL	Аналогично DREADCIC1_CH1_SRC
1:0	DREADCIC1_CH1_SEL	0 - CH1_17 1 - CH2_17 2 - CH3_17 3 - CH4_17

21.2.2.23 Регистры RX_IN_DTRT_TSTx

Назначение: задание тестовых воздействий RX_IN_DTRT_TST1 и RX_IN_DTRT_TST2.

Таблица 21.25. Формат регистра RX_IN_DTRT_TXTx

Номер бита	Условное обозначение	Назначение
31:26	–	Резерв
25:0	TST	Значение тестового воздействия

21.2.2.24 Регистр RX_IN_DTRT_TST_CTRL

Назначение: выбор источника тестового воздействия для PROBE и источника для блока оценки параметров сигнала.

Таблица 21.26. Формат регистра RX_IN_DTRT_TST_CTRL

Номер бита	Условное обозначение	Назначение
31:16	–	Резерв
15	SE_START_STATUS	Запись: единицы инициирует старт процедуры оценки сигнала Чтение: 0 - не готов 1 - оценка завершена
14	–	Резерв
13	PROBE_ENA	Разрешение выхода записи выборки сигнала 1 - разрешено
12:10	PROBE_SRC	Выбор источника для блока записи выборки - см. таблицу в разделе описания PROBE
9:8	TSTPTRN_SEL	Выбор тестового воздействия: 0 - регистры RX_IN_DTRT_TST1 для I компоненты, RX_IN_DTRT_TST2 для Q компоненты; 1 - «пила» для для I компоненты, 0 для Q компоненты; 2 - импульс для для I компоненты, 0 для Q компоненты; 3 - меандр для для I компоненты, 0 для Q компоненты;
7:4	SE_SRC	Выбор источника для Signal Estimator - см. описание Signal Estimator
3:0	SE_EST_DURATION	Длительность оценки параметров сигналов равна $2^{SE\_EST\_DURATION}$

21.2.2.25 Регистр RX_IN_DTRT_OVF_STATUS

Назначение: отображение перегрузки приемных каналов.

Таблица 21.27. Формат регистра RX_IN_DTRT_OVF_STATUS

Номер бита	Условное обозначение	Назначение
31:4	–	Резерв
3	OVF_INP_CH4	Перегрузка входного сигнала CH4
2	OVF_INP_CH3	Перегрузка входного сигнала CH3
1	OVF_INP_CH2	Перегрузка входного сигнала CH2
0	OVF_INP_CH1	Перегрузка входного сигнала CH1

21.2.2.26 Регистры RX_IN_DTRT_INCH_OFFSETx

Назначение: установка значения коррекции смещения нуля в канале x.

Таблица 21.28. Формат регистра RX_IN_DTRT_INCH_OFFSETx

Номер бита	Условное обозначение	Назначение
31:16	–	Резерв
15:0	OFFSET	Коррекция смещения нуля в канале x. Двоично-дополнительное знаковое число.

21.2.2.27 Регистры RX_IN_DTRT_INCH_OVFTRSx

Назначение: задание порога для детектора перегрузки входного сигнала в канале x.

Таблица 21.29. Формат регистра RX_IN_DTRT_INCH_ OVFTRSx

Номер бита	Условное обозначение	Назначение
31:15	–	Резерв
14:0	OFFSET	Порог детектирования перегрузки. Двоично-дополнительное беззнаковое число.

21.2.2.28 Регистр RX_IN_DTRT_SE_POWRMS

Назначение: вывод оценок мощности сигнала RMS

Таблица 21.30. Формат регистра RX_IN_DTRT_SE_POWRMS

Номер бита	Условное обозначение	Назначение
31:16	–	Резерв
15:8	Mant	Мантисса, беззнаковое целое.
7:0	Exp	Экспонента, беззнаковое целое.

21.2.2.29 Регистры RX_IN_DTRT_SE_MIN, RX_IN_DTRT_SE_MAX, RX_IN_DTRT_SE_MEAN

Назначение: вывод максимального, минимального и среднего значений сигнала.

Таблица 21.31. Формат регистра RX_IN_DTRT_INCH_(MIN/MAX/MEAN)

Номер бита	Условное обозначение	Назначение
31:16	–	Резерв
15:0	ESTIMATION	Оценка параметра. Двоично-дополнительное знаковое число.

21.2.2.30 Регистр RX_SUM_CTRL

Назначение: управление сумматором приемных каналов.

Таблица 21.32. Формат регистра RX_SUM_CTRL

Номер бита	Условное обозначение	Назначение
31:16	–	Резерв
15	UPDATE_ACK	Подтверждение обновления коэффициента 1 - означает, что обновление произошло. 0 - означает, что перезапись не состоялась. (RO)
14:10	–	Резерв
9	UPDATE_WR	1 - обновления по факту записи коэффициента разрешены
8	UPDATE_SYNC	1 - обновления по сигналу SYNC_CFG разрешены
7:4	SCALE	Управление устройством масштабирования выхода RX_SUM. Регулировка уровня выходного сигнала RX_SUM: 0: scale[25:0] = out[41:16] 1: scale[25:0] = out[40:15] … 15: scale[25:0] = out[26:1]
3:0	SUM_MASK	Маска суммирования каналов. Бит, установленный в «1» разрешает участие соответствующего канала в формировании суммы

21.2.2.31 Регистры DRE_ADCICx_CONTROL_STATUS

Назначение: управление режимами устройства DREADCIC с номером x.

Таблица 21.33. Формат регистра DRE_ADCICx_CONTROL_STATUS

Номер бита	Условное обозначение	Назначение
31:16	–	Резерв
15	DRE_ADCIC_ENA	Разрешение работы блока 1 - Блок включен 0 - Блок выключен.
14	MODE	Режим работы блока 1 - DRE mode 0 - ADCIC mode
13:9	–	Резерв
8	KOFF_UPDATE	Запись единицы принудительно обновляет актуальные коэффициенты из регистров управления. (WO)
7	EST_STATUS	Чтение: признак окончания вычисления оценки Запись: перезапуск в ручном режиме
6	EST_AUTOREPEAT	1 - процесс оценки автоматически перезапускается 0 - ручной запуск
5	DRE_AUTO_GAIN_ADJ	Сигнал разрешения автоматической коррекции усиления в режиме DRE 1 - разрешено 0 - запрещено
4	–	Резерв

Номер бита	Условное обозначение	Назначение
3:0	EST_PERIOD	Периодичность оценки параметров сигналов Период равен $2^{(EST\_PERIOD+4)}$ отсчетов

21.2.2.32 Регистры DRE_ADCICx_GAIN_ERR

Назначение: чтение оценки ошибки баланса усиления в каналах устройства DREADCIC с номером x в режиме расширения динамического диапазона.

Таблица 21.34. Формат регистров DRE_ADCICx_GAIN_ERR

Номер бита	Условное обозначение	Назначение
31:16	–	Резерв
15:0	GAIN_ERR	Выходной сигнал оценки ошибки усиления

21.2.2.33 Регистры DRE_ADCICx_OFFSET_ERR

Назначение: чтение оценки разности смещения нуля в каналах устройства DREADCIC с номером x в режиме DRE.

Таблица 21.35. Формат регистров DRE_ADCICx_OFFSET_ERR

Номер бита	Условное обозначение	Назначение
31:16	–	Резерв
15:0	OFFSET_ERR	Оценка смещения между каналами

21.2.2.34 Регистры DRE_ADCICx_K1

Назначение: задание коэффициента K1 для устройства DREADCIC с номером x.

Таблица 21.36. Формат регистров DRE_ADCICx_K1

Номер бита	Условное обозначение	Назначение
31:14	–	Резерв
13:0	K1	Двоичное положительное число. В режиме DRE: усиление в канале 1, дробная часть. В режиме ADCIC: добавка усиления в канале A для компенсации амплитудного небаланса

21.2.2.35 Регистры DRE_ADCICx_K2

Назначение: задание коэффициента K2 для устройства DREADCIC с номером x.

Таблица 21.37. Формат регистров DRE_ADCICx_K2

Номер бита	Условное обозначение	Назначение
31:13	–	Резерв
12:0	K2	Двоичное знаковое число. В режиме DRE: усиление в канале 1, целая часть, 8 бит в младших разрядах. В режиме ADCIC: добавка в канал B из канала A для компенсации фазового небаланса

21.2.2.36 Регистры DRE_ADCICx_K3

Назначение: задание коэффициента K3 для устройства DREADCIC с номером x.

Таблица 21.38. Формат регистров DRE_ADCICx_K3

Номер бита	Условное Обозначение	Назначение
31:14	–	Резерв
13:0	K3	Двоичное положительное число В режиме DRE не используется. В режиме ADCIC: добавка усиления в канале B для компенсации амплитудного небаланса

21.2.2.37 Регистры DRE_ADCICx_GAIN_INT

Назначение: чтение актуального значения целочисленной части коэффициента усиления для канала CH1_IN в устройстве с номером x, как результата его автоматической коррекции в режиме DRE.

Таблица 21.39. Формат регистров DRE_ADCICx_GAIN_INT

Номер бита	Условное Обозначение	Назначение
31:16	–	Резерв
15:0	GAIN_INT	Двоичное положительное число. Актуальное значение целой части коэффициента усиления DRE_ADCICx

21.2.2.38 Регистры DRE_ADCICx_GAIN_FRC

Назначение: чтение актуального значения дробной части коэффициента усиления для канала CH1_IN в устройстве с номером x, как результата его автоматической коррекции в режиме DRE.

Таблица 21.40. Формат регистров DRE_ADCICx_GAIN_FRC

Номер бита	Условное Обозначение	Назначение
31:16	–	Резерв
15:0	GAIN_FRC	Двоичное положительное число. Актуальное значение дробной части коэффициента усиления DRE_ADCICx.

21.2.2.39 Регистр MTCFLT_CFG

Назначение: конфигурация согласованного фильтра.

Таблица 21.41. Формат регистра MTCFLT_CFG

Номер бита	Условное обозначение	Назначение
31:9	–	Резерв
8:5	MTCFLT_SCALE	Управление устройством масштабирования выхода MTCFLT128. Регулировка уровня компонент выходного сигнала фильтра: 0 - scale[15:0] = out[31:16] 1 - scale[15:0] = out[30:15] … 15 - scale[15:0] = out[16:1]

Номер бита	Условное обозначение	Назначение
4:0	ORDER	Кол-во коэффициентов, обрабатываемых в каждом из 4 ядер

21.2.2.40 Регистры MTCFLT_COREx_TAPy

Назначение: четыре последовательно расположенных банка регистров для задания коэффициентов четырех вычислительных ядер фильтра. В наименовании регистра номер ядра задан литерой x, заменяемой на 0,1,2,3. Размер каждого банка – 32 коэффициента. Номер коэффициента y=0,1…31.

Таблица 21.42. Формат регистров MTCFLT_COREx_TAPy

Номер бита	Условное обозначение	Назначение
31:16	–	Резерв
15:8	COEFF_I	I - компонента. Целое число со знаком.
4:0	COEFF_Q	Q - компонента. Целое число со знаком.

21.2.2.41 Регистр RX_CHNL_CTRL

Назначение: общая конфигурация приемного канала.

Таблица 21.43. Формат регистра RX_CHNL_CTRL

Номер бита	Условное обозначение	Назначение
31:16	–	Резерв
15	ENABLE	Разрешение работы канала 0 - канал выключен и находится в режиме низкого энергопотребления 1 - канал включен
14	STATE	Состояние канала 0 - канал находится в ожидании SYNC_START 1 - SYNC_START=1 (RO)
13	SYNC_EVENT	Флаг произошедшего события, что строб SYNC_CFG достиг выходного интерфейса (DATA_OUT). Сброс осуществляется записью единицы в этот бит.
12	PPS_EVENT	Флаг произошедшего события, что строб PPS детектирован на входе Сброс осуществляется записью единицы в этот бит.
11:8	–	Резерв
7:0	OVF_STATUS	Статус перегрузок блоков. Запись 1 в бит сбрасывает его. Бит – Примечание OVF_STATUS [0] – DCIC2 OVF_STATUS [1] – DCICN OVF_STATUS [2] – DHB1 OVF_STATUS [3] – DHB2 OVF_STATUS [4] – DHB3 OVF_STATUS [5] – DFIR64 первый OVF_STATUS [6] – DFIR64 второй OVF_STATUS [7] – FGAIN

21.2.2.42 Регистр RX_CHNL_PPS_POS

Назначение: чтение позиции последнего детектированного события PPS относительно старта передачи, выраженного в количестве отсчетов частоты АЦП.

Таблица 21.44. Формат регистра RX_CHNL_PPS_POS

Номер бита	Условное обозначение	Назначение
31:0	PPS_POS	Позиция строба PPS относительно начала входного потока.

21.2.2.43 Регистры RX_NCO_FRQ, TX_NCO_FRQ

Назначение: задание частоты гетеродина в приемном и передающем канале.

Таблица 21.45. Формат регистров RX_NCO_FRQ, TX_NCO_FRQ

Номер бита	Условное Обозначение	Назначение
31:0	NCO_FRQ	Частота гетеродина

21.2.2.44 Регистры RX_NCO_SWRATE, TX_NCO_SWRATE

Назначение: задание скорости свипа частоты гетеродина в приемном и передающем канале.

Таблица 21.46. Формат регистров RX_NCO_SWRATE, TX_NCO_SWRATE

Номер бита	Условное Обозначение	Назначение
31:0	NCO_SWRATE	Приращение частоты гетеродина

21.2.2.45 Регистры RX_NCO_SWTIME, TX_NCO_SWTIME

Назначение: задание периода свипа гетеродина в приемном и передающем канале.

Таблица 21.47. Формат регистров RX_NCO_SWTIME, TX_NCO_SWTIME

Номер бита	Условное Обозначение	Назначение
31:0	NCO_SWTIME	Период свипа в периодах частоты АЦП/ЦАП

21.2.2.46 Регистры RX_NCO_PHASE, TX_NCO_PHASE

Назначение: задание смещения фазы гетеродина в приемном и передающем канале.

Таблица 21.48. Формат регистров RX_NCO_PHASE, TX_NCO_PHASE

Номер бита	Условное обозначение	Назначение
31:0	NCO_PHASE	Смещение фазы гетеродина

21.2.2.47 Регистры RX_NCO_MODE, TX_NCO_MODE

Назначение: управление режимами гетеродина в приемном и передающем канале.

Таблица 21.49. Формат регистров RX_NCO_MODE, TX_NCO_MODE

Номер бита	Условное обозначение	Назначение
31:16	–	Резерв
15	NCO_ENA	Разрешение переноса частоты 0 - off (NCO Bypass) 1 - NCO включен

Номер бита	Условное обозначение	Назначение
14	REG_TRANSPARENT	Прозрачность регистров 0 - off (обновляются только по SYNC_REG) 1 - on прозрачны всегда
13:7	–	Резерв
6:5	SWEEP_MODE	Режим свипа частоты 0 - выключено 1 - SAW 1 2 - SAW 2 3 - SAW 3
4:3	–	Резерв
2	PHASE_SYNC	Включение/выключение синхронизации регистров NCO_PHASE по сигналу SYNC_CFG 0 - off 1 - on
1	FRQ_SYNC	Включение/выключение синхронизации регистров NCO_FRQ по сигналу SYNC_CFG 0 - off 1 - on
0	SWEEP_SYNC	Включение/выключение синхронизации регистров свипа по сигналу SYNC_CFG 0 - off 1 - on

21.2.2.48 Регистр DCIC2_CFG

Назначение: управление конфигурацией устройства DCIC2.

Таблица 21.50. Формат регистра DCIC2_CFG

Номер бита	Условное обозначение	Назначение
31:16	–	Резерв
15	CLR	Принудительный сброс. Читается нулем (WO)
14	–	Резерв
13:8	DCIC2_DR	Коэффициент децимации+1 0 - фильтр выключен. Режим bypass 1 - коэффициент децимации 2 и.т.д
7	DCIC2_ORDERX2	Удвоение порядка фильтра относительно требуемого коэффициента децимации 0 - порядок равен коэффициенту децимации 1 - удвоенный.
6:4	–	Резерв
3:0	DCIC2_SCALE	Управление устройством масштабирования выхода DCIC2. Регулировка уровня выходного сигнала DCIC2: 0: scale[25:0] = out[41:16] 1: scale[25:0] = out[40:15] … 15: scale[25:0] = out[26:1]

21.2.2.49 Регистр DCICN_CFG

Назначение: управление конфигурацией устройства DCICN.

Таблица 21.51. Формат регистра DCICN_CFG

Номер бита	Условное обозначение	Назначение
31:16	–	Резерв
15	CLR	Принудительный сброс. Читается нулем (WO)
14:8	DCICN_DR	Коэффициент децимации+1 0 - каскад выключен. Режим bypass 1 - коэффициент децимации 2 и.т.д
7:6	DCICN_MODE	Биты конфигурации DCICN: 0 - Второй каскад децимации включен в режиме CIC3. 1 - Второй каскад децимации включен в режиме CIC4. 2 - Второй каскад децимации включен в режиме CIC5. 3 - Второй каскад децимации включен в режиме CIC6.
5	–	Резерв
4	DCICN_SCALE_RNG	Управление дополнительным масштабированием На вход SCALER подаются старшие биты с выхода DCICN Подаются младшие биты
3:0	DCICN_SCALE	Управление устройством масштабирования выхода DCICN. Регулировка уровня выходного сигнала DCICN: 0: scale[25:0] = out[41:16] 1: scale[25:0] = out[40:15] … 15: scale[25:0] = out[26:1]

21.2.2.50 Регистр DCIC_BWGAUGE

Назначение: чтение результатов оценки степени использования разрядности шины на выходе DCIC2 и DCICN.

Таблица 21.52. Формат регистра DCIC_BWGAUGE

Номер бита	Условное обозначение	Назначение
31:10	–	Резерв
9:5	DCIC2_BWGAUGE	Оценка уровня сигнала по выходу DCIC2
4:0	DCICN_BWGAUGE	Оценка уровня сигнала по выходу DCICN

21.2.2.51 Регистр DHB_MODE

Назначение: управление конфигурацией устройства DHB.

Таблица 21.53. Формат регистра DHB_MODE

Номер бита	Условное обозначение	Назначение
31:16	–	Резерв
15	CLR	Запись единицы осуществляет сброс всех стадий
14	–	Резерв
13:12	DHB_MODE	Конфигурация DHB: 0 - фильтры и децимация DHB выключены (bypass) 1 - DHB3 включен, DHB1 и DHB2 выключены, децимация 1:2 2 - DHB3, DHB2 включены, DHB1 выключен, децимация 1:4 3 - DHB3, DHB2, DHB1 включены децимация 1:8
11	–	Резерв

Номер бита	Условное обозначение	Назначение
10:8	DHB3_SCALE	Управление устройством масштабирования выхода DHB3. Регулировка уровня выходного сигнала DHB3: 0: scale[25:0] = out[41:16] 1: scale[25:0] = out[40:15] … 7: scale[25:0] = out[34:9]
7	–	Резерв
6:4	DHB2_SCALE	Управление устройством масштабирования выхода DHB2. Регулировка уровня выходного сигнала DHB2: 0: scale[25:0] = out[41:16] 1: scale[25:0] = out[40:15] … 7: scale[25:0] = out[34:9]
3	–	Резерв
2:0	DHB1_SCALE	Управление устройством масштабирования выхода DHB1. Регулировка уровня выходного сигнала DHB1: 0: scale[25:0] = out[41:16] 1: scale[25:0] = out[40:15] … 7: scale[25:0] = out[34:9]

21.2.2.52 Регистр DHB_BWGAUGE

Назначение: чтение результатов оценки степени использования разрядности шины на выходе DHB.

Таблица 21.54. Формат регистра DHB_BWGAUGE

Номер бита	Условное обозначение	Назначение
31:15	–	Резерв
14:10	DHB1_ BWGAUGE	Оценка уровня сигнала по выходу DHB1
9:5	DHB2_ BWGAUGE	Оценка уровня сигнала по выходу DHB2
4:0	DHB3_ BWGAUGE	Оценка уровня сигнала по выходу DHB3

21.2.2.53 Регистр RX_FGAIN_COFF

Назначение: установка комплексного коэффициента масштабирования и поворота фазы выхода приемного канала.

Таблица 21.55. Формат регистра RX_FGAIN_COFF

Номер бита	Условное обозначение	Назначение
31:16	COFF_RE	COFF_RE реальная часть коэффициента усиления в двоично-дополнительном коде. Значение 8192 соответствует единичному коэффициенту передачи (1+0*j). Значение по сбросу: 8192.
15:0	COFF_IM	COFF_IM - мнимая часть коэффициента усиления в двоично-дополнительном коде. Значение 8192 соответствует единичному коэффициенту передачи (0+1*j). Значение по сбросу: 0.

21.2.2.54 Регистр RX_FGAIN_CTRL

Назначение: управление устройством масштабирования и поворота фазы.

Таблица 21.56. Формат регистра RX_FGAIN_CTRL

Номер бита	Условное обозначение	Назначение
31:16	–	Резерв
15	ENABLE	Включение/выключение блока 1 - рабочий режим. 0 - режим bypass.
14:11	–	Резерв
10	UPDATE_ACK	Подтверждение обновления коэффициента 1 - означает, что обновление произошло. 0 - означает, что перезапись не состоялась. (RO)
9	UPDATE_WR	Разрешение обновления по факту записи коэффициента
8	UPDATE_SYNC	Разрешение обновления по сигналу SYNC_CFG
7:5	–	Резерв
4:0	BWGAUGE	Оценка уровня сигнала по выходу фильтра.

21.2.2.55 Регистры DFIRx_Py_CFk

Назначение: группа регистров для записи коэффициентов фильтров DFIR64. В названии литера x соответствует номеру устройства (0,1), литера y – номеру профиля (0,1), литера k – номеру коэффициента фильтра (0,1..31).

Коэффициент (целое число со знаком) записывается в младшие 24 бита регистра, при чтении возвращается 32-битное значение, расширенное знаком.

21.2.2.56 Регистры DFIRx_PROFILEy

Назначение: группа регистров для задания характеристик профиля фильтра DFIR64. В наименовании регистров DFIRx_PROFILEy литера x – номер устройства, литера y – номер профиля.

Таблица 21.57. Формат регистров DFIRx_PROFILEy

Номер бита	Условное Обозначение	Назначение
31:16	–	Резерв
15:14	SYM	Симметричность: 0 - симметричный, порядок до 64 1,3 - несимметричный. Порядок до 32 2 - антисимметричный, порядок до 64
13:8	ORDER	Порядок фильтра: число отводов = ORDER + 1
7	–	Резерв
6:4	DR	Уменьшенный на 1 коэффициент децимации
3:0	SCALE	Управление устройством масштабирования выхода DFIR64. Регулировка уровня выходного сигнала DFIR64: 0: scale[25:0] = out[54:29] 1: scale[25:0] = out[53:28] … 15: scale[25:0] = out[39:14]

21.2.2.57 Регистр DFIRx_CFG

Назначение: управление включением, режимами конфигурации и переключение активного профиля DFIR с номером x.

Таблица 21.58. Формат регистра DFIRx_CFG

Номер бита	Условное Обозначение	Назначение
31:16	–	Резерв
15	ENA	0 - каскад выключен. Режим bypass 1 - каскад включен
14:8	–	Резерв
7	PRF_TRANSP	0 - Регистры управления непрозрачны 1 - Регистры управления прозрачны
6	SYNCCFGSENCE	0 - Обновление профиля по SYNC_CFG запрещено. 1 - Обновление профиля по SYNC_CFG разрешено
5:1	–	Резерв
0	PROFILE	Активный профиль

21.2.2.58 Регистр DFIRx_ BWGAUGE

Назначение: чтение результатов оценки степени использования разрядности шины на выходе DFIRx.

Таблица 21.59. Формат регистра DFIRx_ BWGAUGE

Номер бита	Условное Обозначение	Назначение
31: 5	–	Резерв
4:0	BWGAUGE	Оценка уровня сигнала по выходу фильтра.

21.2.2.59 Регистр TX_CTRL

Назначение: выбор режима записи, номера канала (страницы адресации регистров) при конфигурации передающих каналов.

Таблица 21.60. Формат регистра TX_CTRL

Номер бита	Условное Обозначение	Назначение
31:3	–	Резерв
2	CFG_WR_ALL	Управление записью конфигурации одновременно во все каналы. 1 - Запись всех каналов. 0 - Запись канала, определяемого CH_CFG_SEL
1:0	CH_CFG_SEL	Выбор канала для записи/чтения регистров управления при страничной адресации 0 - первый канал 1 - второй канал 2,3 - зарезервировано

21.2.2.60 Регистр DAC_CTRL

Назначение: управление режимами ЦАП и преобразованием формата сигнала.

Таблица 21.61. Формат регистра DAC_CTRL

Номер бита	Условное Обозначение	Назначение
31:16	–	Резерв
15:14	DST_SEL	Глобальное включение/выключение интерфейса, выбор порта. 0 - интерфейс выключен 1 - 16 бит параллельная шина CMOS порта 2 - 16 бит параллельная шина LVDS порта 3 - 4 канала JESD204 интерфейса
13:11	PBUS_MODE	0 - Simple Parallel SDR 1 - Simple Parallel DDR мультиплексирование каналов 2 - Simple Parallel DDR мультиплексирование бит 3 - Simple Parallel DDR мультиплексирование отсчетов 4 - JESD207 SDR (AD9361 CMOS) 5 - JESD207 DDR (AD9361 CMOS) 6 - AD9361 DDR LVDS (DDR) 7 - Reserved (Simple Parallel SDR)
10	MODE_OPT1	Разрядность шины данных ЦАП для SRC_SEL=1,2 при PBUS_MODE=0, 1, 3 0 - 16 бит 1 - 8 бит Порядок следования данных PBUS_MODE=2 0 - Первым следуют младшие (четные) биты. (по активному фронту DAC_CLK) 1 - Первым следуют старшие (нечетные) биты. (по активному фронту DAC_CLK) Количество каналов для SRC_SEL=1,2 при PBUS_MODE =4, 5, 6 0 - 1 канал 1 - 2 канала
9:8	MODE_OPT2	Разрядность шины данных ЦАП для PBUS_MODE=6. 0 - 8 бит 1 - 7 бит 2 - 6 бит 3 - 5 бит
7	CLK_ERR	Признак ошибки синхронизации перехода между тактовыми доменами
6	FRAME_AL	Активный уровень сигнала фреймовой синхронизации JESD207.
5	PUSH	Запись: 1 - старт процедуры PUSH, 0 - игнорируется Чтение: 1 - окончание процедуры PUSH
4:1	–	Резерв
0	TC2OB_ON	Включение/выключение преобразования кода из двоично-дополнительного в прямо-смещенный 0 - off 1 - on

21.2.2.61 Регистр TX_OUT_DTRT_ROUTE_CTRL

Назначение: управление коммутацией и выравниванием потоков в передатчике

Таблица 21.62. Формат регистра TX_OUT_DTRT_ROUTE_CTRL

Номер бита	Условное Обозначение	Назначение
31:4	–	Резерв
3	CH_ALIGN_ENA	Разрешение ”выравнивания каналов” по сигналу PUSH.
2	IQ_SWAP	Обмен I и Q компонент. 0 - не менять 1 - менять
1:0	ROUTE_MODE	Выбор режима трассировки потоков. См. таблицу в описании

21.2.2.62 Регистр TX_OUT_DTRT_OVF_STATUS

Назначение: индикация перегрузки выходных каналов передающего тракта

Таблица 21.63. Формат регистра TX_OUT_DTRT_OVF_STATUS

Номер бита	Условное Обозначение	Назначение
31:4	–	Резерв
3	OVF_CH4	Перегрузка выходного сигнала CH4
2	OVF_CH3	Перегрузка выходного сигнала CH3
1	OVF_CH2	Перегрузка выходного сигнала CH2
0	OVF_CH1	Перегрузка выходного сигнала CH1

21.2.2.63 Регистры TX_OUT_DTRT_TSTx

Назначение: задание тестовых воздействий TX_OUT_DTRT_TST1 и TX_OUT_DTRT_TST2.

Таблица 21.64. Формат регистра TX_OUT_DTRT_TXTx

Номер бита	Условное обозначение	Назначение
31:18	–	Резерв
17:0	TST	Значение тестового воздействия

21.2.2.64 Регистр TX_OUT_DTRT_CHx_OFFSET

Назначение: задание коррекции смещения нуля в канале x.

Таблица 21.65. Формат регистра TX_OUT_DTRT_ CH1x_OFFSET

Номер бита	Условное обозначение	Назначение
31:16	–	Резерв
15:0	OFFSET	Значение коррекции смещения

21.2.2.65 Регистр TX_OUT_DTRT_TST_CTRL

Назначение: выбор источника сигнала для PROBE и управление блоком оценки параметров сигнала.

Таблица 21.66. Формат регистра TX_OUT_DTRT_TST_CTRL

Номер бита	Условное обозначение	Назначение
31:16	–	Резерв
15	SE_START_STATUS	Запись единицы инициирует старт процедуры оценки сигнала Чтение: 0 - не готов 1 - оценка завершена
14	–	Резерв
13	PROBE_ENA	Разрешение выхода записи выборки сигнала
12	–	Резерв

Номер бита	Условное обозначение	Назначение
11:10	PROBE_SRC	Выбор источника для блока записи выборки. 0 - DAC_CH2 (в старших 16 битах) DAC_CH1 (в младших) 1 - DAC_CH3 (в старших 16 битах) DAC_CH4 (в младших)
9:6	–	Резерв
5:4	SE_SRC	Выбор источника для Signal Estimator. 0 - DAC_CH1 (знакорасширенный) 1 - DAC_CH2 (знакорасширенный) 2 - DAC_CH3 (знакорасширенный) 3 - DAC_CH4 (знакорасширенный)
3:0	SE_EST_DURATION	Длительность оценки параметров сигналов Период 2^ SE_EST_DURATION

21.2.2.66 Регистр TX_OUT_DTRT_SE_POWRMS

Назначение: вывод результатов оценки мощности.

Таблица 21.67. Формат регистра TX_OUT_DTRT_SE_POWRMS

Номер бита	Условное обозначение	Назначение
31:16	–	Резерв
15:8	MANT	мантисса
7:0	EXP	экспонента

21.2.2.67 Регистры TX_OUT_DTRT_SE_MIN, TX_OUT_DTRT_SE_MAX, TX_OUT_DTRT_SE_MEAN

Назначение: вывод результатов оценок максимального, минимального и среднего значений сигнала.

Таблица 21.68. Формат регистра TX_IN_DTRT_INCH_(MIN/MAX/MEAN)

Номер бита	Условное обозначение	Назначение
31:16	–	Резерв
15:0	ESTIMATION	Оценка параметра. Двоично-дополнительное знаковое число.

21.2.2.68 Регистр TX_SUM_CTRL

Назначение: управление сумматором передающих каналов.

Таблица 21.69. Формат регистра TX_SUM_CTRL

Номер бита	Условное обозначение	Назначение
31:16	–	Резерв
15	UPDATE_ACK	Подтверждение обновления коэффициента 1 - означает, что обновление произошло. 0 - означает, что перезапись не состоялась. (RO)
14:10	–	Резерв
9	UPDATE_WR	Запись единицы в поле UPDATE_WR приводит к немедленному применению (обновлению маски) значением SUM_MASK.
8	UPDATE_SYNC	Разрешение обновления маски по сигналу SYNC_CFG
7:2	–	Резерв

Номер бита	Условное обозначение	Назначение
1:0	SUM_MASK	Маска суммирования каналов

21.2.2.69 Регистр TX_CHNL_CTRL

Назначение: общая конфигурация передающего канала.

Таблица 21.70. Формат регистра TX_CHNL_CTRL

Номер бита	Условное обозначение	Назначение
31:16	–	Резерв
15	ENABLE	Разрешение работы канала 0 - канал выключен и находится в режиме низкого энергопотребления 1 - канал включен
14	STATE	Состояние канала 0 - канал находится в ожидании SYNC_START 1 - SYNC_START=1 (RO)
13	SYNC_EVENT	Флаг произошедшего события, что строб SYNC_CFG достиг выходного интерфейса (DATA_OUT). Сброс осуществляется записью единицы в этот бит.
12	PPS_EVENT	Флаг произошедшего события, что строб PPS детектирован на входе Сброс осуществляется записью единицы в этот бит.
11	PM_OUT	Состояние сигнала PUSH_MARK_OUT (RO)
9	INVSINC_ENA	Разрешение работы INVSINC фильтра 0 - байпас фильтра 1 - фильтр включен.
8	MODE	Режим работы 0 - Без планировщика задач 1 - Под управлением планировщика задач.
7	–	Резерв
6:0	OVF_STATUS	Статус перегрузок блоков. Запись 1 в бит сбрасывает его. Бит — Примечание OVF_STATUS [0] — CIC2 OVF_STATUS [1] — CICN OVF_STATUS [2] — IHB1 OVF_STATUS [3] — IHB2 OVF_STATUS [4] — IHB3 OVF_STATUS [5] — DFIR64 первый OVF_STATUS [6] — DFIR64 второй

21.2.2.70 Регистр TX_CHNL_TASK_STATE

Назначение: управление планировщиком старта передачи.

Таблица 21.71. Формат регистра TX_CHNL_TASK_STATE

Номер бита	Условное обозначение	Назначение
31:16	–	Резерв
15	SCH_ENABLE	Разрешение работы планировщика. Ноль временно останавливает работу планировщика.
14	QUEUE_RESET	Сброс очереди задач. Запись единицы сбрасывает очередь.

Номер бита	Условное обозначение	Назначение
13:8		
7:5	PPS_CNT	Счетчик PPS.
4:0	TASK_CNT	Количество задач в очереди. При чтении возвращается кол-во занятых ячеек в очереди. 0 соответствует пустой очереди

21.2.2.71 Регистры TX_CHNL_DT1, TX_CHNL_DT2

Назначение: Задание защитных временных интервалов перед стартом передачи фрейма (DT1) и после окончания передачи (DT2).

Таблица 21.72. Формат регистра TX_CHNL_DT1(2)

Номер бита	Условное обозначение	Назначение
31:24	–	Резерв
23:0	DT	Длительность защитного интервала в отсчетах ЦАП

21.2.2.72 Регистр TX_CHNL_TASK_TME

Назначение: добавление задачи в очередь планировщика старта передачи.

Таблица 21.73. Формат регистра TX_CHNL_TASK_TME

Номер бита	Условное обозначение	Назначение
31:0	TASK_TME	Время задержки старта новой передачи относительно PPS.

21.2.2.73 Регистр ICIC2_CFG

Назначение: управление конфигурацией устройства ICIC2.

Таблица 21.74. Формат регистра ICIC2_CFG

Номер бита	Условное обозначение	Назначение
31:16	–	Резерв
15	CLR	Принудительный сброс. Читается нулем (WO)
14:8	ICIC2_IR	Коэффициент интерполяции, уменьшенный на 1 0 - каскад выключен. Режим bypass 1 - коэффициент интерполяции 2 и.т.д
7	ICIC2_ORDERX2	Удвоение порядка фильтра относительно требуемого коэффициента интерполяции 0 - порядок равен коэффициенту интерполяции 1 - порядок удвоенный
6:4	–	Резерв
3:0	ICIC2_SCALE	Управление устройством масштабирования выхода ICIC2. Регулировка уровня выходного сигнала ICIC2: 0: scale[17:0] = out[26:9] 1: scale[17:0] = out[25:8] … 9: scale[17:0] = out[17:0] 10: scale[17:0] = {out[16:0],1'b0} 11: scale[17:0] = {out[15:0],2'b0} …

Номер бита	Условное обозначение	Назначение
		15: scale[17:0] = {out[11:0],6'b0}

21.2.2.74 Регистр ICICN_CFG

Формат регистра ICICN_CFG приведен в следующей таблице:

Таблица 21.75. Формат регистра ICICN_CFG

Номер бита	Условное обозначение	Назначение
31:16	–	Резерв
15	CLR	Принудительный сброс. Читается нулем (WO)
14:8	ICICN_IR	Коэффициент интерполяции-1 0 - каскад выключен. Режим bypass 1 - коэффициент интерполяции 2 и.т.д
7:6	ICICN_MODE	Количество задействованных каскадов: 0 - CIC3 1 - CIC4 2 - CIC5 3 - CIC6
5	–	Резерв
4	ICICN_SCALE_RNG	Управление дополнительным масштабированием 0 - на вход SCALER подаются старшие 34 бита с выхода ICICN 1 - на вход SCALER подаются младшие 34 бита с выхода ICICN
3:0	ICICN_SCALE	Управление устройством масштабирования выхода ICICN. Регулировка уровня выходного сигнала ICICN: 0: scale[17:0] = out[33:16] 1: scale[17:0] = out[32:15] … 15: scale[17:0] = out[18:1]

21.2.2.75 Регистр ICIC_BWGAUGE

Назначение: чтение результатов оценки степени использования разрядности шины на выходе ICIC2 и ICICN.

Таблица 21.76. Формат регистра ICIC_BWGAUGE

Номер бита	Условное обозначение	Назначение
31:10	–	Резерв
9:5	ICIC2_BWGAUGE	Оценка уровня сигнала по выходу ICIC2
4:0	ICICN_BWGAUGE	Оценка уровня сигнала по выходу ICICN

21.2.2.76 Регистр IHB_MODE

Назначение: управление конфигурацией устройства IHB.

Таблица 21.77. Формат регистра IHB_MODE

Номер бита	Условное Обозначение	Назначение
31:16	–	Резерв

Номер бита	Условное Обозначение	Назначение
15	CLR	Запись единицы осуществляет сброс всех стадий
14	–	Резерв
13:12	IHB_MODE	Биты конфигурации IHB: 0 - Каскад HB выключен (bypass) 1 - Каскад HB включен в режиме интерполяции в 2 раза стадией IHB3 2 - Каскад HB включен в режиме интерполяции в 4 раза стадией IHB2+IHB3 3 - Каскад HB включен в режиме интерполяции в 8 раз тремя стадиями IHB1+ IHB2+IHB3
11:0	–	Резерв

21.2.2.77 Регистр IHB_BWGAUGE

Назначение: чтение результатов оценки степени использования разрядности шины на выходе IHB.

Таблица 21.78. Формат регистра IHB_BWGAUGE

Номер бита	Условное Обозначение	Назначение
31:15	–	Резерв
14:10	IHB3_BWGAUGE	Оценка уровня сигнала по выходу IHB3
9:5	IHB2_BWGAUGE	Оценка уровня сигнала по выходу IHB2
4:0	IHB1_BWGAUGE	Оценка уровня сигнала по выходу IHB1

21.2.2.78 Регистр MOD_MODE

Назначение: управление расширением спектра и режимом модуляции сигнала.

Таблица 21.79. Формат регистра MOD_MODE

Номер бита	Условное Обозначение	Назначение
31:2	–	Резерв
1:0	MODE	Режим работы: 0 - прямая передача данных в квадратуры (DIRECT_DATA) 1 - преобразование по таблице (MEMORY_MAP, QAM) 2 - прямое расширение PN последовательностью символьного потока 3 - прямое расширение PN последовательностью битового потока

21.2.2.79 Регистр MOD_PN_PROFILE

Назначение: управление модулятором в режиме прямого расширения спектра сигнала PN последовательностью.

Таблица 21.80. Формат регистра MOD_PN_PROFILE

Номер бита	Условное обозначение	Назначение
31:16	–	Резерв
15:8	SEQ_LENGTH	Длина PN кода минус 1
7:4	PROFILE_I	Профиль для I канала
3:0	PROFILE_Q	Профиль для Q канала

21.2.2.80 Регистры IFIRx_Py_CFk

Назначение: группа регистров для записи коэффициентов фильтров IFIR64. В названии литера x соответствует номеру устройства (0,1), литера y – номеру профиля (0,1), литера k – номеру коэффициента фильтра (0,1..31).

Коэффициент (целое число со знаком) записывается в младшие 24 бита регистра, при чтении возвращается 32-битное значение, расширенное знаком.

21.2.2.81 Регистры IFIRx_PROFILEy

Назначение: группа регистров для задания характеристик профиля фильтра IFIR64. В наименовании регистров IFIRx_PROFILEy литера x – номер устройства, литера y – номер профиля.

Таблица 21.81. Формат регистры IFIR_PROFILE0, IFIR_PROFILE1

Номер бита	Условное Обозначение	Назначение
31:16	–	Резерв
15:14	SYM	Симметричность: 0 - симметричный, порядок до 64 1,3 - несимметричный. Порядок до 32 2 - антисимметричный, порядок до 64
13:8	ORDER	Порядок фильтра: число отводов = ORDER + 1
7	INTERP_MODE	Режим интерполяции 0 - прореживание нулевыми отсчетами 1 - повторение отсчетов
6:4	IR	Уменьшенный на 1 коэффициент интерполяции
3:0	SCALE	Управление устройством масштабирования выхода DFIR64. Регулировка уровня выходного сигнала DFIR64: 0: scale[17:0] = out[46:29] 1: scale[17:0] = out[45:28] … 15: scale[17:0] = out[31:14]

21.2.2.82 Регистр IFIRx_CFG

Назначение: управление включением, режимами конфигурации и переключение активного профиля IFIR с номером x.

Таблица 21.82. Формат регистра IFIRx_CFG

Номер бита	Условное обозначение	Назначение
31:16	–	Резерв
15	ENA	0 - фильтр выключен. Режим bypass 1 - фильтр включен
14:8	–	Резерв
7	PRF_TRANSP	0 - регистры управления непрозрачны 1 - регистры управления прозрачны
6	SYNCCFGSENCE	0 - обновление профиля по SYNC_CFG запрещено. 1 - обновление профиля по SYNC_CFG разрешено
5:1	–	Резерв
0	PROFILE	Активный профиль

21.2.2.83 Регистр IFIRx_ BWGAUGE

Назначение: чтение результатов оценки степени использования разрядности шины на выходе IFIRx.

Таблица 21.83. Формат регистра IFIR0(1)_ BWGAUGE

Номер бита	Условное обозначение	Назначение
31:5	–	Резерв
4:0	BWGAUGE	Оценка уровня сигнала по выходу фильтра.

21.2.2.84 Регистры MOD_MEMx

Назначение: диапазон регистров для доступа в память модулятора.

Литера x в названии равна смещению 32-битного регистра от начала диапазона и может принимать значения 0..255. Для записи и чтения памяти используются только младшие 16 бит регистров.

Память организована как 16 строк длиной 256 бит каждая. Младшие биты номера регистра x[3:0] выбирают номер 16 разрядного слова в 256 битной строке, начиная с младших битов. Старшие биты x[7:4] задают номер 256 битной строки.

21.2.3 Регистры блока DFE_NELS (общий доступ - APB)

Таблица 21.84. Перечень программно-доступных регистров

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
DFE_commander0	DFE Commander блока 0	0x0	RW	0x0
DFE_comm0_read_first	DFE Commander блока 0, чтение FIFO GPIO, тэг, читать первым	0x0	RO	0x4
DFE_commander0_status	DFE Commander блока 0, чтение статуса.	0x0	RO	0x8
DDUC_Control0	DDUC блока 0: настройка тактовых сигналов и прерываний	0x0	WO	0xc
DFE_commander1	DFE Commander блока 1	0x0	RW	0x10
DFE_comm1_read_first	DFE Commander блока 1, чтение FIFO GPIO, тэг, читать первым	0x0	RO	0x14
DFE_commander1_status	DFE Commander блока 1, чтение статуса.	0x0	RO	0x18
DDUC_Control1	DDUC блока 1: настройка тактовых сигналов и прерываний	0x0	WO	0x1c
DFE_commander2	DFE Commander блока 2	0x0	RW	0x20
DFE_comm2_read_first	DFE Commander блока 2, чтение FIFO GPIO, тэг, читать первым	0x0	RO	0x24
DFE_commander2_status	DFE Commander блока 2, чтение статуса.	0x0	RO	0x28
DDUC_Control2	DDUC блока 2: настройка тактовых сигналов и прерываний	0x0	WO	0x2c
DFE_commander3	DFE Commander блока 3	0x0	RW	0x30
DFE_comm3_read_first	DFE Commander блока 3, чтение FIFO GPIO, тэг, читать первым	0x0	RO	0x34
DFE_commander3_status	DFE Commander блока 3, чтение статуса.	0x0	RO	0x38

DDUC_Control3	DDUC блока 3: настройка тактовых сигналов и прерываний	0x0	WO	0x3c
Attenuator0_ch0_settings	Настройки аттенюатора	0x0	WO	0x80
Attenuator0_ch0_data0	UP_EXCEEDS and UP_PERIOD	0x0	WO	0x84
Attenuator0_ch0_data1	SETTLE_TIME AND DOWN_PERIOD	0x0	WO	0x88
Attenuator0_ch0_uplevel	UP_LEVEL	0x0	WO	0x8c
Attenuator1_ch0_settings	Настройки аттенюатора и чтение текущего значения	0x0	RW	0x90
Attenuator1_ch0_data0	UP_EXCEEDS and UP_PERIOD	0x0	WO	0x94
Attenuator1_ch0_data1	SETTLE_TIME AND DOWN_PERIOD	0x0	WO	0x98
Attenuator1_ch0_uplevel	UP_LEVEL	0x0	WO	0x9c
Attenuator0_ch1_settings	Настройки аттенюатора и чтение текущего значения	0x0	RW	0xa0
Attenuator0_ch1_data0	UP_EXCEEDS and UP_PERIOD	0x0	WO	0xa4
Attenuator0_ch1_data1	SETTLE_TIME AND DOWN_PERIOD	0x0	WO	0xa8
Attenuator0_ch1_uplevel	UP_LEVEL	0x0	WO	0xac
Attenuator1_ch1_settings	Настройки аттенюатора и чтение текущего значения	0x0	RW	0xb0
Attenuator1_ch1_data0	UP_EXCEEDS and UP_PERIOD	0x0	WO	0xb4
Attenuator1_ch1_data1	SETTLE_TIME AND DOWN_PERIOD	0x0	WO	0xb8
Attenuator1_ch1_uplevel	UP_LEVEL	0x0	WO	0xbc
Attenuator0_ch2_settings	Настройки аттенюатора и чтение текущего значения	0x0	RW	0xc0
Attenuator0_ch2_data0	UP_EXCEEDS and UP_PERIOD	0x0	WO	0xc4
Attenuator0_ch2_data1	SETTLE_TIME AND DOWN_PERIOD	0x0	WO	0xc8
Attenuator0_ch2_uplevel	UP_LEVEL	0x0	WO	0xcc
Attenuator1_ch2_settings	Настройки аттенюатора и чтение текущего значения	0x0	RW	0xd0
Attenuator1_ch2_data0	UP_EXCEEDS and UP_PERIOD	0x0	WO	0xd4
Attenuator1_ch2_data1	SETTLE_TIME AND DOWN_PERIOD	0x0	WO	0xd8
Attenuator1_ch2_uplevel	UP_LEVEL	0x0	WO	0xdc
Attenuator0_ch3_settings	Настройки аттенюатора и чтение текущего значения	0x0	RW	0xe0
Attenuator0_ch3_data0	UP_EXCEEDS and UP_PERIOD	0x0	WO	0xe4
Attenuator0_ch3_data1	SETTLE_TIME AND DOWN_PERIOD	0x0	WO	0xe8
Attenuator0_ch3_uplevel	UP_LEVEL	0x0	WO	0xec
Attenuator1_ch3_settings	Настройки аттенюатора и чтение текущего значения	0x0	RW	0xf0
Attenuator1_ch3_data0	UP_EXCEEDS and UP_PERIOD	0x0	WO	0xf4
Attenuator1_ch3_data1	SETTLE_TIME AND DOWN_PERIOD	0x0	WO	0xf8
Attenuator1_ch3_uplevel	UP_LEVEL	0x0	WO	0xfc
DAC1_MUX	DAC1 MUX	0x0	WO	0x100
DAC2_MUX	DAC2 MUX	0x0	WO	0x104
DAC3_MUX	DAC3 MUX	0x0	WO	0x108
TX_RFD_MUX_Sel	TX RFD MUX Sel	0x0	WO	0x10c
TX_Sum	TX Channel Summators	0x0	WO	0x110
ADC_Demapper_mode	Режимы ADC Type Demapper, ADC1-3 MUX	0x0	WO	0x114
ADC_Demapper_conf	ADC Demapper configuration	0x0	WO	0x118
Crypto_module	Криптомодуль: запись кодового слова	0x0	WO	0x11c

21.2.3.1 Регистр DFE_commander

Формат регистра DFE_commander(0-3) приведен в следующей таблице:

Таблица 21.85. Формат регистра DFE_commander

Номер бита	Условное обозначение	Назначение
31:24	Command	Команда
23:0	Data	Данные для команды

21.2.3.2 Регистр DFE_comm_read_first

Формат регистра DFE_comm(0-3)_read_first приведен в следующей таблице:

Таблица 21.86. Формат регистра DFE_comm_read_first

Номер бита	Условное обозначение	Назначение
31:0	FIFO_GPIO	

21.2.3.3 Регистр DFE_commander_status

Формат регистра DFE_commander(0-3)_status приведен в следующей таблице:

Таблица 21.87. Формат регистра DFE_commander_status

Номер бита	Условное обозначение	Назначение
31:5	–	Резерв
4	GPIO_Empty	очередь данных GPIO пуста
3	–	Резерв
2	GPIO_OVERFLOW	переполнение очереди данных GPIO
1	QUEUE_OVERFLOW	переполнение очереди команд
0	WRONG_COMMAND	неверный код команды

21.2.3.4 Регистр DDUC_Control

Формат регистра DDUC_Control(0-3) приведен в следующей таблице:

Таблица 21.88. Формат регистра DDUC_Control

Номер бита	Условное обозначение	Назначение
31:20	-	Резерв
19:15	clk_game_en	clk enable: 19 бит отвечает за передатчик, биты 15-18 за каналы 0-3 приемника
14:0	irq_mask	маска прерываний dduc

21.2.3.5 Регистр Attenuator_ch_settings

Для каждого из каналов ch(0-3) существует по два регистра настроек аттенюатора. Формат регистров Attenuator(0-1)_ch(0-3)_settings приведен в следующей таблице:

Таблица 21.89. Формат регистра Attenuator_ch_settings

Номер бита	Условное обозначение	Назначение
31:20	-	Резерв
19:16	CS_SPI	SPI Chip Select
15:13	SPI_Speed	
12:11	Mode	00 – одиночный, 01 – парный ведущий, 10 – парный ведомый
10:9	Rough_estimator_type	тип грубого аттенюатора: "00" - 0/10/20/30, "01" - 0/6/12/18, "1X" - грубый аттенюатор отсутствует)
8	Mode	0 - Ручной режим, 1 – Авто
7:1	Manual_or_minimal	ручное или минимальное значение аттенюатора
0	-	Резерв

21.2.3.6 Регистр Attenuator_ch_data0

Для каждого из каналов ch(0-3) существует по два регистра data0 аттенюатора. Формат регистров Attenuator(0-1)_ch(0-3)_data0 приведен в следующей таблице:

Таблица 21.90. Формат регистра Attenuator_ch_data0

Номер бита	Условное обозначение	Назначение
31:16	UP_PERIOD	интервал опроса на повышение аттенюации, 16 бит в тиках период_данных_АЦП*128
15:0	UP_EXCEEDS	количество превышений порога UP_LEVEL для повышения аттенюации

21.2.3.7 Регистр Attenuator_ch_data1

Для каждого из каналов ch(0-3) существует по два регистра data1 аттенюатора. Формат регистров Attenuator(0-1)_ch(0-3)_data1 приведен в следующей таблице:

Таблица 21.91. Формат регистра Attenuator_ch_data1

Номер бита	Условное обозначение	Назначение
31:16	SETTLE_TIME	задержка на время отработки SPI и внешнего аттенюатора, 16 бит в тиках период_данных_АЦП*128
15:0	DOWN_PERIOD	интервал опроса на понижение аттенюации, 16 бит в тиках период_данных_АЦП*128

21.2.3.8 Регистр Attenuator_ch_uplevel

Для каждого из каналов ch(0-3) существует по два регистра uplevel аттенюатора. Формат регистров Attenuator(0-1)_ch(0-3)_uplevel приведен в следующей таблице:

Таблица 21.92. Формат регистра Attenuator_ch_uplevel

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:0	UP_LEVEL	порог сравнения на повышение аттенюации

21.2.3.9 Регистр DAC1_MUX

Формат регистра DAC1_MUX приведен в следующей таблице:

Таблица 21.93. Формат регистра DAC1_MUX

Номер бита	Условное обозначение	Назначение
31:6	-	Резерв
5	SDR_DDR	SDR/DDR
4	nclr	nclr
3:2	TX_I	TX_I канал
1:0	TX_Q	TX_Q канал

21.2.3.10 Регистр DAC2_MUX

Формат регистра DAC2_MUX приведен в следующей таблице:

Таблица 21.94. Формат регистра DAC2_MUX

Номер бита	Условное обозначение	Назначение
31:9	-	Резерв
8	nclr	nclr
7:6	LANE 4	LANE 4 канал
5:4	LANE 3	LANE 3 канал
3:2	LANE 2	LANE 2 канал
1:0	LANE 1	LANE 1 канал

21.2.3.11 Регистр DAC3_MUX

Формат регистра DAC3_MUX приведен в следующей таблице:

Таблица 21.95. Формат регистра DAC3_MUX

Номер бита	Условное обозначение	Назначение
31:6	-	Резерв
5	SDR_DDR	SDR/DDR
4	nclr	nclr
3:2	TX_I	TX_I канал
1:0	TX_Q	TX_Q канал

21.2.3.12 Регистр TX_RFD_MUX_Sel

Формат регистра TX_RFD_MUX_Sel приведен в следующей таблице:

Таблица 21.96. Формат регистра TX_RFD_MUX_Sel

Номер бита	Условное обозначение	Назначение
31:8	-	Резерв
7:6	TX_4	4 канал
5:4	TX_3	3 канал
3:2	TX_2	2 канал
1:0	TX_1	1 канал

21.2.3.13 Регистр TX_Sum

Формат регистра TX_Sum приведен в следующей таблице:

Таблица 21.97. Формат регистра TX_Sum

Номер бита	Условное обозначение	Назначение
31:3	-	Резерв
2	DUC4_MUX_SEL	DUC4_MUX_SEL
1:0	DUC1_MUX_SEL	DUC1_MUS_SEL

21.2.3.14 Регистр ADC_Demapper_mode

Формат регистра ADC_Demapper_mode приведен в следующей таблице:

Таблица 21.98. Формат регистра ADC_Demapper_mode

Номер бита	Условное обозначение	Назначение
31:29	-	Резерв
28:27	Nclr_fifo_5_4	Сброс фифо 5 и 4
26	Nclr_fifo_3	Сброс фифо 3
25:24	Nclr_fifo_2_1	Сброс фифо 2 и 1
23:21	Ch8_mux_control	Управление мультиплексором канала 8
20:18	Ch7_mux_control	Управление мультиплексором канала 7
17:15	Ch6_mux_control	Управление мультиплексором канала 6
14:12	Ch5_mux_control	Управление мультиплексором канала 5
11:9	Ch4_mux_control	Управление мультиплексором канала 4
8:6	Ch3_mux_control	Управление мультиплексором канала 3
5:3	Ch2_mux_control	Управление мультиплексором канала 2
2:0	Ch1_mux_control	Управление мультиплексором канала 1

21.2.3.15 Регистр ADC_Demapper_conf

Формат регистра ADC_Demapper_conf приведен в следующей таблице:

Таблица 21.99. Формат регистра ADC_Demapper_conf

Номер бита	Условное обозначение	Назначение
31:17	-	Резерв
16:15	ADC_4_5_control	Управление АЦП4/5 00 - ADC3544 SDR (adc4_1, adc4_2) 01 - ADC3544 DDR (adc4_1, adc4_2) 10 - ADC3644 DDR1 (adc4_1, adc4_2) 11 - ADC3644 DDR2 (adc4_1, adc4_2)
14	XOR_4_5_enable	Включает xor с нулевым битом для АЦП4-5
13	ADC_1_2_mode	Включает раздельный(0)/сдвоенный(1) режим для АЦП1-2
12:11	ADC_1_control	Управляет демаппингом АЦП1 в раздельном режиме 00 - АЦП LTC2262 (adc1_1) 01 - АЦП AD9434 DDR (adc1_1)

Номер бита	Условное обозначение	Назначение
		10 - АЦП ADC3424 (adc1_1..4)
10:9	ADC_2_control	Управляет демаппингом АЦП2 в раздельном режиме 00 - АЦП LTC2262 (adc2_1) 01 - АЦП AD9434 DDR (adc2_1) 10 - АЦП ADC3424 (adc2_1..4)
8:6	ADC_1_2_control	Управляет демаппингом АЦП1-2 в сдвоенном режиме 000 - АЦП AD9434/AD9684 режим 1 (adc1_1, adc2_1) 001 - АЦП AD9434/AD9684 режим 2 (adc1_1, adc2_1) 010 - АЦП AD9434/AD9684 режим 3 (adc1_1, adc2_1) 011 - АЦП AD9434/AD9684 режим 4 (adc1_1, adc2_1) 100 - АЦП AD9650 LVDS DDR 2x16 (adc1_1, adc2_1) 101 - АЦП ADS4249/AD9684 (adc1_1, adc1_2, adc2_1, adc2_2)
5:3	ADC1_1_zero_control	Обнуление adc1_1 000 – отключить обнуление 001 – обнулить 2 младших бита 010 – обнулить 4 младших бита 011 – обнулить 6 младших бит 100 – обнулить 8 младших бит
2:0	ADC2_1_zero_control	Обнуление adc2_1 000 – отключить обнуление 001 – обнулить 2 младших бита 010 – обнулить 4 младших бита 011 – обнулить 6 младших бит 100 – обнулить 8 младших бит

21.2.3.16 Регистр Crypto_module

Формат регистра Crypto_module приведен в следующей таблице:

Таблица 21.100. Формат регистра Crypto_module

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:0	code_word	Кодовое слово для криптомодуля. Предотвращает нежелательную передачу данных. Варианты кодового слова: 0xC4D7 - в данных разрешено только нечетное число единиц, 0x3B23 - в данных разрешено попеременно четное и нечетное число единиц, 0xEB23 - защита отключена, разрешена передача любых данных Иные кодовые слова - передача любых данных запрещена

21.2.4 Регистры блока DFE_NELS (доступ через DFE Commander)

Таблица 21.101. Регистры блока DFE_NELS

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
DDS RX FREQ ch 0	Регистр кода частоты dds.	0x0	WO	0x0000
DDS RX PHASE ch 0	Регистр кода фазы dds.	0x0	WO	0x0001
DDS RX FREQ SHIFT ch 0	Регистр кода изменения частоты.	0x0	WO	0x0002
Resampler 1 RX coefs ch 0	Resampler 1 rx загрузка коэффициентов, см. документацию	0x0	WO	0x0010
Resampler 1 RX length ch 0	Resampler 1 RX длина фильтра	0x0	WO	0x0011
Resampler 1 RX m_factor ch 0	Resampler 1 RX степень интерполяции	0x0	WO	0x0012
Resampler 1 RX d_factor ch 0	Resampler 1 RX степень децимации	0x0	WO	0x0013
Resampler 2 RX coefs ch 0	Resampler 2 rx загрузка коэффициентов, см. документацию	0x0	WO	0x0014
Resampler 2 RX length ch 0	Resampler 2 RX длина фильтра	0x0	WO	0x0015
Resampler 2 RX m_factor ch 0	Resampler 2 RX степень интерполяции	0x0	WO	0x0016
Resampler 2 RX d_factor ch 0	Resampler 2 RX степень децимации	0x0	WO	0x0017
CIC RX ch 0	CIC фильтр, децимация	0x0	WO	0x0018
CIC RX ch 0 gain	CIC фильтр, усиление	0x0	WO	0x0019
Farrow low RX ch 0	код инкремента фазы ресемплинга, младшие биты	0x0	WO	0x001A
Farrow high RX ch 0	код инкремента фазы ресемплинга, старшие биты	0x0	WO	0x001B
Control reg ch 0	Регистр управления	0x0	WO	0x001C
DDS RX FREQ ch 1	Регистр кода частоты dds.	0x0	WO	0x0020
DDS RX PHASE ch 1	Регистр кода фазы dds.	0x0	WO	0x0021
DDS RX FREQ SHIFT ch 1	Регистр кода изменения частоты.	0x0	WO	0x0022
Resampler 1 RX coefs ch 1	Resampler 1 rx загрузка коэффициентов, см. документацию	0x0	WO	0x0030
Resampler 1 RX length ch 1	Resampler 1 RX длина фильтра	0x0	WO	0x0031
Resampler 1 RX m_factor ch 1	Resampler 1 RX степень интерполяции	0x0	WO	0x0032
Resampler 1 RX d_factor ch 1	Resampler 1 RX степень децимации	0x0	WO	0x0033
Resampler 2 RX coefs ch 1	Resampler 2 rx загрузка коэффициентов, см. документацию	0x0	WO	0x0034
Resampler 2 RX length ch 1	Resampler 2 RX длина фильтра	0x0	WO	0x0035
Resampler 2 RX m_factor ch 1	Resampler 2 RX степень интерполяции	0x0	WO	0x0036
Resampler 2 RX d_factor ch 1	Resampler 2 RX степень децимации	0x0	WO	0x0037
CIC RX ch 1	CIC фильтр, децимация	0x0	WO	0x0038
CIC RX ch 1 gain	CIC фильтр, усиление	0x0	WO	0x0039
Farrow low RX ch 1	код инкремента фазы ресемплинга, младшие биты	0x0	WO	0x003A
Farrow high RX ch 1	код инкремента фазы ресемплинга, старшие биты	0x0	WO	0x003B
Control reg ch 1	Регистр управления	0x0	WO	0x003C

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
DDS RX FREQ ch 2	Регистр кода частоты dds.	0x0	WO	0x0040
DDS RX PHASE ch 2	Регистр кода фазы dds.	0x0	WO	0x0041
DDS RX FREQ SHIFT ch 2	Регистр кода изменения частоты.	0x0	WO	0x0042
Resampler 1 RX coefs ch 2	Resampler 1 rx загрузка коэффициентов, см. документацию	0x0	WO	0x0050
Resampler 1 RX length ch 2	Resampler 1 RX длина фильтра	0x0	WO	0x0051
Resampler 1 RX m_factor ch 2	Resampler 1 RX степень интерполяции	0x0	WO	0x0052
Resampler 1 RX d_factor ch 2	Resampler 1 RX степень децимации	0x0	WO	0x0053
Resampler 2 RX coefs ch 2	Resampler 2 rx загрузка коэффициентов, см. документацию	0x0	WO	0x0054
Resampler 2 RX length ch 2	Resampler 2 RX длина фильтра	0x0	WO	0x0055
Resampler 2 RX m_factor ch 2	Resampler 2 RX степень интерполяции	0x0	WO	0x0056
Resampler 2 RX d_factor ch 2	Resampler 2 RX степень децимации	0x0	WO	0x0057
CIC RX ch 2	CIC фильтр, децимация	0x0	WO	0x0058
CIC RX ch 2 gain	CIC фильтр, усиление	0x0	WO	0x0059
Farrow low RX ch 2	код инкремента фазы ресемплинга, младшие биты	0x0	WO	0x005A
Farrow high RX ch 2	код инкремента фазы ресемплинга, старшие биты	0x0	WO	0x005B
Control reg ch 2	Регистр управления	0x0	WO	0x005C
DDS RX FREQ ch 3	Регистр кода частоты dds.	0x0	WO	0x0060
DDS RX PHASE ch 3	Регистр кода фазы dds.	0x0	WO	0x0061
DDS RX FREQ SHIFT ch 3	Регистр кода изменения частоты.	0x0	WO	0x0062
Resampler 1 RX coefs ch 3	Resampler 1 rx загрузка коэффициентов, см. документацию	0x0	WO	0x0070
Resampler 1 RX length ch 3	Resampler 1 RX длина фильтра	0x0	WO	0x0071
Resampler 1 RX m_factor ch 3	Resampler 1 RX степень интерполяции	0x0	WO	0x0072
Resampler 1 RX d_factor ch 3	Resampler 1 RX степень децимации	0x0	WO	0x0073
Resampler 2 RX coefs ch 3	Resampler 2 rx загрузка коэффициентов, см. документацию	0x0	WO	0x0074
Resampler 2 RX length ch 3	Resampler 2 RX длина фильтра	0x0	WO	0x0075
Resampler 2 RX m_factor ch 3	Resampler 2 RX степень интерполяции	0x0	WO	0x0076
Resampler 2 RX d_factor ch 3	Resampler 2 RX степень децимации	0x0	WO	0x0077
CIC RX ch 3	CIC фильтр, децимация	0x0	WO	0x0078
CIC RX ch 3 gain	CIC фильтр, усиление	0x0	WO	0x0079
Farrow low RX ch 3	код инкремента фазы ресемплинга, младшие биты	0x0	WO	0x007A
Farrow high RX ch 3	код инкремента фазы ресемплинга, старшие биты	0x0	WO	0x007B
Control reg ch 3	Регистр управления	0x0	WO	0x007C
ADC DDC bypass	ADC DDC bypass control	0x0	?	0x0080
Zero offset correction I RX	zero offset correction I control	0x0	RW	0x0100

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
Zero offset correction Q RX	zero offset correction Q control	0x0	RW	0x0101
Angle correction RX	Angle correction control	0x0	WO	0x0102
Gain correction I RX	Gain correction I control	0x0	WO	0x0103
Gain correction Q RX	Gain correction Q control	0x0	WO	0x0104
Dither I RX	Dither I control	0x0	WO	0x0105
Dither Q RX	Dither Q control	0x0	WO	0x0106
Timer	Регистр настройки таймера	0x0	WO	0x0180
DDS TX FREQ	Регистр кода частоты dds.	0x0	WO	0x0400
DDS TX PHASE	Регистр кода фазы dds.	0x0	WO	0x0401
DDS TX FREQ SHIFT	Регистр кода изменения частоты.	0x0	WO	0x0402
Resampler 1 TX coefs	Resampler 1 rx загрузка коэффициентов, см. документацию	0x0	WO	0x0410
Resampler 1 TX length	Resampler 1 RX длина фильтра	0x0	WO	0x0411
Resampler 1 TX m_factor	Resampler 1 RX степень интерполяции	0x0	WO	0x0412
Resampler 1 TX d_factor	Resampler 1 RX степень децимации	0x0	WO	0x0413
Resampler 2 TX coefs	Resampler 2 rx загрузка коэффициентов, см. документацию	0x0	WO	0x0414
Resampler 2 TX length	Resampler 2 RX длина фильтра	0x0	WO	0x0415
Resampler 2 TX m_factor	Resampler 2 RX степень интерполяции	0x0	WO	0x0416
Resampler 2 TX d_factor	Resampler 2 RX степень децимации	0x0	WO	0x0417
CIC TX	CIC фильтр, децимация	0x0	WO	0x0418
CIC TX gain	CIC фильтр, усиление	0x0	WO	0x0419
Farrow TX Low	код инкремента фазы ресемплинга, младшие биты	0x0	WO	0x041A
Farrow TX High	код инкремента фазы ресемплинга, старшие биты	0x0	WO	0x041B
Control reg TX	Регистр управления duc	0x0	WO	0x041C
FIFO Threshold TX	Регистр границ заполнения FIFO	0x0	WO	0x041D
Angle correction TX	Angle correction control	0x0	WO	0x0440
Gain correction I TX	Gain correction I control	0x0	WO	0x0441
Gain correction Q TX	Gain correction Q control	0x0	WO	0x0442
Zero offset correction I TX	zero offset correction I control	0x0	WO	0x0443
Zero offset correction Q TX	zero offset correction Q control	0x0	WO	0x0444
IQ UP DDS TX FREQ	Регистр кода частоты dds.	0x0	WO	0x0450
IQ UP DDS TX PHASE	Регистр кода фазы dds.	0x0	WO	0x0451
IQ UP DDS TX FREQ SHIFT	Регистр кода изменения частоты.	0x0	WO	0x0452
IQ UP MIXER Control	IQ UP MIXER управление	0x0	WO	0x0458
IQ UpSampling коэф-ы fir	адреса 0x480-0x493 прямой доступ к памяти	0x0	WO	0x0480 - 0x0493
IQ UpSampling Control	IQ UpSampling управление	0x0	WO	0x0494
TX CFR FIR	адреса 0x4C0-0x4C5 TX Crest Factor Reduction коэф-ы fir-фильтра	0x0	WO	0x04C0 - 0x04C5
TX CFR Control	TX Crest Factor Reduction управление	0x0	WO	0x04CF
TX CFR lookup	адреса 0x4D0-0x4DF TX Crest Factor Reduction коэф-ы lookup table	0x0	WO	0x04D0 - 0x04DF
TX Digital Predistortion Amp	Адреса 0x500-0x57F TX Digital Predistortion Таблица аппроксимации для коррекции амплитуды	0x0	WO	0x0500 - 0x057F
TX Digital Predistortion Ph	Адреса 0x580-0x5FF TX Digital Predistortion Таблица аппроксимации для коррекции фазы	0x0	WO	0x0580 - 0x05FF

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
TX Digital Predistortion Control	TX Digital Predistortion регистр управления	0x0	WO	0x04E0

21.2.4.1 Регистр «DDS RX FREQ»

Формат регистра «DDS RX FREQ (ch0 – ch3)» приведен в следующей таблице:

Таблица 21.102. Формат регистра «DDS RX FREQ»

Номер бита	Условное обозначение	Назначение
31:0	freq	код частоты

21.2.4.2 Регистр «DDS RX PHASE»

Формат регистра «DDS RX PHASE (ch0 – ch3)» приведен в следующей таблице:

Таблица 21.103. Формат регистра «DDS RX PHASE»

Номер бита	Условное обозначение	Назначение
31:0	phase	код фазы

21.2.4.3 Регистр «DDS RX FREQ SHIFT»

Формат регистра «DDS RX FREQ SHIFT (ch0 – ch3)» приведен в следующей таблице:

Таблица 21.104. Формат регистра «DDS RX FREQ SHIFT»

Номер бита	Условное обозначение	Назначение
31:0	freq_shift	код изменения частоты

21.2.4.4 Регистр «Resampler RX coefs»

Формат регистра «Resampler (1, 2) RX coefs (ch0 – ch3)» приведен в следующей таблице:

Таблица 21.105. Формат регистра «Resampler RX coefs»

Номер бита	Условное обозначение	Назначение
31:24	-	Резерв
23	run	флаг начала загрузки
22:18	-	Резерв
17:0	coefs	Коэффициенты фильтра

21.2.4.5 Регистр «Resampler RX length»

Формат регистра «Resampler (1, 2) RX length (ch0 – ch3)» приведен в следующей таблице:

Таблица 21.106. Формат регистра «Resampler RX length»

Номер бита	Условное обозначение	Назначение
31:9	-	Резерв
8:0	length	Длина фильтра

21.2.4.6 Регистр «Resampler RX m_factor»

Формат регистра «Resampler (1, 2) RX m_factor (ch0 – ch3)» приведен в следующей таблице:

Таблица 21.107. Формат регистра «Resampler RX m_factor»

Номер бита	Условное обозначение	Назначение
31:6	-	Резерв
5:0	m_factor	Степень интерполяции

21.2.4.7 Регистр «Resampler RX d_factor»

Формат регистра «Resampler (1, 2) RX d_factor (ch0 – ch3)» приведен в следующей таблице:

Таблица 21.108. Формат регистра «Resampler RX d_factor»

Номер бита	Условное обозначение	Назначение
31:6	-	Резерв
5:0	d_factor	Степень децимации

21.2.4.8 Регистр «CIC RX»

Формат регистра «CIC RX (ch0 – ch3)» приведен в следующей таблице:

Таблица 21.109. Формат регистра «CIC RX»

Номер бита	Условное обозначение	Назначение
31:14	-	Резерв
13:0	cic	Регистр степени децимации (задавать как R-1)

21.2.4.9 Регистр «CIC RX gain»

Формат регистра «CIC RX gain (ch0 – ch3)» приведен в следующей таблице:

Таблица 21.110. Формат регистра «CIC RX gain»

Номер бита	Условное обозначение	Назначение
31:6	-	Резерв
5:0	cic_gain	Регистр усиления (к-во сдвигов после интегрирования)

21.2.4.10 Регистр «Farrow low RX»

Формат регистра «Farrow low RX (ch0 – ch3)» приведен в следующей таблице:

Таблица 21.111. Формат регистра «Farrow low RX»

Номер бита	Условное обозначение	Назначение
31:0	farrow_low	Код инкремента фазы ресэмплинга, биты (31:0)

21.2.4.11 Регистр «Farrow high RX»

Формат регистра «Farrow high RX (ch0 – ch3)» приведен в следующей таблице:

Таблица 21.112. Формат регистра «Farrow high RX»

Номер бита	Условное обозначение	Назначение
31:0	farrow_high	Код инкремента фазы ресэмплинга, биты (63:32)

21.2.4.12 Регистр «Control reg»

Формат регистра «Control reg (ch0 – ch3)» приведен в следующей таблице:

Таблица 21.113. Формат регистра «Control reg»

Номер бита	Условное Обозначение	Назначение
31:5	-	Резерв
4	Mode	режим входа DDC, '0’ – одиночный, '1' – квадратурный, IQ
3:0	Select	выбор канала ADC входа I

21.2.4.13 Регистр «ADC DDC Bypass»

Формат регистра «ADC DDC Bypass» приведен в следующей таблице:

Таблица 21.114. Формат регистра «ADC DDC Bypass»

Номер бита	Условное Обозначение	Назначение
31:5	-	Резерв
4	clr_run	DDC Bypass CLR/RUN
3	-	Резерв
2:0	Select	DDC bypass выбор канала ADC

21.2.4.14 Регистр «Zero offset correction I RX»

Формат регистра «Zero offset correction I RX» приведен в следующей таблице:

Таблица 21.115. Формат регистра «Zero offset correction I RX»

Номер бита	Условное Обозначение	Назначение
31:24	-	Резерв
23	Bypass	пропуск данных через блок без изменения
22	nclr_out	очистка
21	nclr_hold	очистка регистра коррекции
20	run_hold	остановка/запуск блока
19:18	-	Резерв
17:16	t_const	0 - стандартная скорость 1 - ускорение в 8 раз 2 - ускорение в 64 раза 3 - ускорение в 256 раз
15:0	zero_correction	коррекция нулей

21.2.4.15 Регистр «Zero offset correction Q RX»

Формат регистра «Zero offset correction Q RX» приведен в следующей таблице:

Таблица 21.116. Формат регистра «Zero offset correction Q RX»

Номер бита	Условное Обозначение	Назначение
31:24	-	Резерв
23	bypass	пропуск данных через блок без изменения
22	nclr_out	очистка
21	nclr_hold	очистка регистра коррекции
20	run_hold	остановка/запуск блока
19:18	-	Резерв
17:16	t_const	0 - стандартная скорость 1 - ускорение в 8 раз 2 - ускорение в 64 раза 3 - ускорение в 256 раз
15:0	zero_correction	коррекция нулей

21.2.4.16 Регистр «Angle correction RX»

Формат регистра «Angle correction RX» приведен в следующей таблице:

Таблица 21.117. Формат регистра «Angle correction RX»

Номер бита	Условное обозначение	Назначение
31:24	-	Резерв
23	bypass	пропуск данных через блок без изменения
22	nclr_out	очистка блока
21	-	Резерв
20	run_hold	старт/стоп блока
19:16	-	Резерв
15:0	sin(a)	синус угла поворота

21.2.4.17 Регистр «Gain correction I RX»

Формат регистра «Gain correction I RX» приведен в следующей таблице:

Таблица 21.118. Формат регистра «Gain correction I RX»

Номер бита	Условное обозначение	Назначение
31:24	-	Резерв
23	bypass	
22	nclr_out	
21	-	Резерв
20	run_hold	
19:16	-	Резерв
15:0	gain	

21.2.4.18 Регистр «Gain correction Q RX»

Формат регистра «Gain correction Q RX» приведен в следующей таблице:

Таблица 21.119. Формат регистра «Gain correction Q RX»

Номер бита	Условное обозначение	Назначение
31:24	-	Резерв
23	bypass	
22	nclr_out	
21	-	Резерв
20	run_hold	
19:16	-	Резерв
15:0	gain	

21.2.4.19 Регистр «Dither I RX»

Формат регистра «Dither I RX» приведен в следующей таблице:

Таблица 21.120. Формат регистра «Dither I RX»

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:0	dither	

21.2.4.20 Регистр «Dither Q RX»

Формат регистра «Dither Q RX» приведен в следующей таблице:

Таблица 21.121. Формат регистра «Dither Q RX»

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:0	dither	

21.2.4.21 Регистр «Timer»

Формат регистра «Timer» приведен в следующей таблице:

Таблица 21.122. Формат регистра «Timer».

Номер бита	Условное обозначение	Назначение
31:5	-	Резерв
4	nclr	
3	clk_out_enable	управление выходным портом DFE_timer_oe двунаправленного порта DFE_timer_clk
2:0	divider	делитель частоты, с которой переключается счетчик (1/2/4/8/16/32/64).

21.2.4.22 Регистр «DDS TX FREQ»

Формат регистра «DDS TX FREQ» приведен в следующей таблице:

Таблица 21.123. Формат регистра «DDS TX FREQ»

Номер бита	Условное обозначение	Назначение
31:0	freq	код частоты

21.2.4.23 Регистр «DDS TX PHASE»

Формат регистра «DDS TX PHASE» приведен в следующей таблице:

Таблица 21.124. Формат регистра «DDS TX PHASE»

Номер бита	Условное обозначение	Назначение
31:0	phase	код фазы

21.2.4.24 Регистр «DDS TX FREQ SHIFT»

Формат регистра «DDS TX FREQ SHIFT» приведен в следующей таблице:

Таблица 21.125. Формат регистра «DDS TX FREQ SHIFT»

Номер бита	Условное обозначение	Назначение
31:0	freq_shift	код изменения частоты

21.2.4.25 Регистр «Resampler TX coefs»

Формат регистра «Resampler (1, 2) TX coefs» приведен в следующей таблице:

Таблица 21.126. Формат регистра «Resampler TX coefs»

Номер бита	Условное обозначение	Назначение
31:24	-	Резерв
23	run	флаг начала загрузки
22:18	-	Резерв
17:0	coefs	Коэффициенты фильтра

21.2.4.26 Регистр «Resampler TX length»

Формат регистра «Resampler (1, 2) TX length» приведен в следующей таблице:

Таблица 21.127. Формат регистра «Resampler TX length»

Номер бита	Условное обозначение	Назначение
31:9	-	Резерв
8:0	length	Длина фильтра

21.2.4.27 Регистр «Resampler TX m_factor»

Формат регистра «Resampler (1, 2) TX m_factor» приведен в следующей таблице:

Таблица 21.128. Формат регистра «Resampler TX m_factor»

Номер бита	Условное обозначение	Назначение
31:6	-	Резерв
5:0	m_factor	Степень интерполяции

21.2.4.28 Регистр «Resampler TX d_factor»

Формат регистра «Resampler (1, 2) TX d_factor» приведен в следующей таблице:

Таблица 21.129. Формат регистра «Resampler TX d_factor»

Номер бита	Условное обозначение	Назначение
31:6	-	Резерв
5:0	d_factor	Степень децимации

21.2.4.29 Регистр «CIC TX»

Формат регистра «CIC TX» приведен в следующей таблице:

Таблица 21.130. Формат регистра «CIC TX»

Номер бита	Условное обозначение	Назначение
31:14	-	Резерв
13:0	cic	Регистр степени децимации (задавать как R-1)

21.2.4.30 Регистр «CIC TX gain»

Формат регистра «CIC TX gain» приведен в следующей таблице:

Таблица 21.131. Формат регистра «CIC TX gain»

Номер бита	Условное Обозначение	Назначение
31:6	-	Резерв
5:0	cic_gain	Регистр усиления (к-во сдвигов после интегрирования)

21.2.4.31 Регистр «Farrow low TX»

Формат регистра «Farrow low TX» приведен в следующей таблице:

Таблица 21.132. Формат регистра «Farrow low TX»

Номер бита	Условное Обозначение	Назначение
31:0	farrow_low	Код инкремента фазы ресэмплинга, биты (31:0)

21.2.4.32 Регистр «Farrow high TX»

Формат регистра «Farrow high TX» приведен в следующей таблице:

Таблица 21.133. Формат регистра «Farrow high TX»

Номер бита	Условное Обозначение	Назначение
31:0	farrow_high	Код инкремента фазы ресэмплинга, биты (63:32)

21.2.4.33 Регистр «Control reg TX»

Формат регистра «Control reg TX» приведен в следующей таблице:

Таблица 21.134. Формат регистра «Control reg TX»

Номер бита	Условное Обозначение	Назначение
31:3	-	Резерв

Номер бита	Условное Обозначение	Назначение
2	Mode	режим выхода: ‘0’ – IQ, '1' – комплексное умножение на DDS
1	Load	Загрузка DDS из временных регистров
0	Nclr	nclr FIFO

21.2.4.34 Регистр «FIFO Threshold TX»

Формат регистра «FIFO Threshold TX» приведен в следующей таблице:

Таблица 21.135. Формат регистра «FIFO Threshold TX»

Номер бита	Условное Обозначение	Назначение
31:16	dma_high_threshild	Верхняя граница FIFO
15:0	dma_low_threshild	Нижняя граница FIFO

21.2.4.35 Регистр «Angle correction TX»

Формат регистра «Angle correction TX» приведен в следующей таблице:

Таблица 21.136. Формат регистра «Angle correction TX»

Номер бита	Условное Обозначение	Назначение
31:24	-	Резерв
23	Bypass	пропуск данных через блок без изменения
22	nclr_out	очистка блока
21	-	Резерв
20	run_hold	старт/стоп блока
19:16	-	Резерв
15:0	sin(a)	синус угла поворота

21.2.4.36 Регистр «Gain correction I TX»

Формат регистра «Gain correction I TX» приведен в следующей таблице:

Таблица 21.137. Формат регистра «Gain correction I TX»

Номер бита	Условное обозначение	Назначение
31:24	-	Резерв
23	bypass	
22	nclr_out	
21	-	Резерв
20	run_hold	
19:18	-	Резерв
17:0	gain	

21.2.4.37 Регистр «Gain correction Q TX»

Формат регистра «Gain correction Q TX» приведен в следующей таблице:

Таблица 21.138. Формат регистра «Gain correction Q TX»

Номер бита	Условное обозначение	Назначение
31:24	-	Резерв

Номер бита	Условное обозначение	Назначение
23	bypass	
22	nclr_out	
21	-	Резерв
20	run_hold	
19:18	-	Резерв
17:0	gain	

21.2.4.38 Регистр «Zero offset correction I TX»

Формат регистра «Zero offset correction I TX» приведен в следующей таблице:

Таблица 21.139. Формат регистра «Zero offset correction I TX»

Номер бита	Условное обозначение	Назначение
31:24	-	Резерв
23	bypass	пропуск данных через блок без изменения
22	nclr_out	очистка
21	-	Резерв
20	run_hold	остановка/запуск блока
19:16	-	Резерв
15:0	zero_correction	коррекция нулей

21.2.4.39 Регистр «Zero offset correction Q TX»

Формат регистра «Zero offset correction Q TX» приведен в следующей таблице:

Таблица 21.140. Формат регистра «Zero offset correction Q TX»

Номер бита	Условное обозначение	Назначение
31:24	-	Резерв
23	bypass	пропуск данных через блок без изменения
22	nclr_out	очистка
21	-	Резерв
20	run_hold	остановка/запуск блока
19:16	-	Резерв
15:0	zero_correction	коррекция нулей

21.2.4.40 Регистр «IQ UP DDS TX FREQ»

Формат регистра «IQ UP DDS TX FREQ» приведен в следующей таблице:

Таблица 21.141. Формат регистра «IQ UP DDS TX FREQ»

Номер бита	Условное обозначение	Назначение
31:0	freq	код частоты

21.2.4.41 Регистр «IQ UP DDS TX PHASE»

Формат регистра «IQ UP DDS TX PHASE» приведен в следующей таблице:

Таблица 21.142. Формат регистра «IQ UP DDS TX PHASE»

Номер бита	Условное Обозначение	Назначение
31:0	Phase	код фазы

21.2.4.42 Регистр «IQ UP DDS TX FREQ SHIFT»

Формат регистра «IQ UP DDS TX FREQ SHIFT» приведен в следующей таблице:

Таблица 21.143. Формат регистра «IQ UP DDS TX FREQ SHIFT»

Номер бита	Условное Обозначение	Назначение
31:0	freq_shift	код изменения частоты

21.2.4.43 Регистр «IQ UP MIXER Control»

Формат регистра «IQ UP MIXER Control» приведен в следующей таблице:

Таблица 21.144. Формат регистра «IQ UP MIXER Control»

Номер бита	Условное Обозначение	Назначение
31:2	-	Резерв
1	Load	Загрузка DDS из временных регистров
0	Nclr	

21.2.4.44 Регистр «IQ UpSampling FIR Coefs»

Формат регистра «IQ UpSampling FIR Coefs(0-19)» приведен в следующей таблице:

Таблица 21.145. Формат регистра «IQ UpSampling FIR Coefs»

Номер бита	Условное Обозначение	Назначение
31:0	Coefs	Коэффициенты FIR

21.2.4.45 Регистр «IQ UPSampling Control»

Формат регистра «IQ UPSampling Control» приведен в следующей таблице:

Таблица 21.146. Формат регистра «IQ UPSampling Control»

Номер бита	Условное обозначение	Назначение
31:2	-	Резерв
1	bypass	
0	nclr	

21.2.4.46 Регистр «TX CFR FIR»

Формат регистра «TX CFR FIR(0-5)» приведен в следующей таблице:

Таблица 21.147. Формат регистра «TX CFR FIR»

Номер бита	Условное обозначение	Назначение
31:0	coef	запись коэффициентов FIR-фильтра

21.2.4.47 Регистр «TX CFR Control»

Формат регистра «TX CFR Control» приведен в следующей таблице:

Таблица 21.148. Формат регистра «TX CFR Control»

Номер бита	Условное обозначение	Назначение
31:24	-	Резерв
23	bypass	
22	nclr	
21	mult_enable	
20	run_hold	
19	lookup_switch	смена рабочей таблицы
18:16	-	Резерв
15:0	mult_value	

21.2.4.48 Регистр «TX CFR lookup»

Формат регистра «TX CFR lookup (0-15)» приведен в следующей таблице:

Таблица 21.149. Формат регистра «TX CFR lookup»

Номер бита	Условное обозначение	Назначение
31:0	coef	запись таблицы lookup

21.2.4.49 Регистр «TX Digital Predistortion Amp»

Формат регистра «TX Digital Predistortion Amp (0-127)» приведен в следующей таблице:

Таблица 21.150. Формат регистра «TX Digital Predistortion Amp»

Номер бита	Условное обозначение	Назначение
31:0	coef	Таблица аппроксимации для коррекции амплитуды

21.2.4.50 Регистр «TX Digital Predistortion Ph»

Формат регистра «TX Digital Predistortion Ph (0-127)» приведен в следующей таблице:

Таблица 21.151. Формат регистра «TX Digital Predistortion Ph»

Номер бита	Условное обозначение	Назначение
31:0	coef	Таблица аппроксимации для коррекции фазы

21.2.4.51 Регистр «TX Digital Predistortion Control»

Формат регистра «TX Digital Predistortion Control» приведен в следующей таблице:

Таблица 21.152. Формат регистра «TX Digital Predistortion Control»

Номер бита	Условное обозначение	Назначение
31:24	-	Резерв
23	bypass	
22	nclr	
21	-	Резерв
20	run_hold	
19	TABLE_SWITCH_K	смена рабочей таблицы для коррекции амплитуды
18	TABLE_SWITCH_F	смена рабочей таблицы для коррекции фазы
17:0	-	Резерв

21.3 Функционирование

21.3.1 DFE_ALT

Основное назначение блока – выделение узкополосных сигналов из входного широкополосного сигнала, поступающего из внешнего АЦП (приемный тракт) и формирование высокочастотного сигнала для подачи на внешний ЦАП (передающий тракт). Блок может работать в непрерывном или пакетном режимах передачи. Помимо этого, в блоке осуществляются вспомогательные операции, связанные со взаимной синхронизацией принимаемых и передаваемых сигналов и устройств обработки. Для синхронизации работы ПО с DFE_ALT формируются сигналы прерываний.

21.3.1.1 Синхронизация приемных каналов

DFE_ALT обладает возможностью синхронизации старта и смены конфигурации каналов обработки приемного тракта. Управление формированием сигналов синхронизации осуществляется через поля регистра SYNC_CTRL. Первый фрейм данных, переданный через DMA после старта приемника, начинается с отсчета, пришедшего на вход интерфейса АЦП в момент активизации сигнала RX_SYNC_START.

Сигнал синхронизации старта работы приемных каналов RX_SYNC_START формируется в домене тактовой частоты дискретизации (АЦП). Событием, приводящим к старту обработки в приемных каналах, является нарастающий фронт сигнала PPS или сигнал от ПО, переданный как запись «1» в поле SYNCSTARTOUT регистра SYNC_CTRL. Поле RX_START_PPSSYNC определяет режим генерации сигнала синхронизации старта.

В режиме без синхронизации старта по PPS (RX_START_PPSSYNC=0): сигнал RX_SYNC_START формируется по первому же активному фронту тактовой частоты домена АЦП, возникающему после изменения состояния поля SYNCSTARTOUT регистра SYNC_CTRL с «0» на «1».

В режиме с синхронизацией старта по PPS (RX_START_PPSSYNC=1): запись единицы в поле SYNCSTARTOUT регистра SYNC_CTRL не приводит к формированию сигнала старта, но разрешает его формирование по сигналу PPS. Сигнал RX_SYNC_START формируется по первому же активному фронту тактовой частоты АЦП, возникающему после нарастающего фронта входного сигнала PPS, асинхронного по отношению к тактовому домену АЦП. Сброс сигнала в ноль осуществляется без привязки к PPS при записи в поле SYNCSTARTOUT значения «0».

При непрерывной работе приемника в каждом приемном канале производится подсчет принятых отсчетов на 32-разрядном счетчике, работающем с переполнением. В момент возникновения события PPS значение счетчика копируется в регистр RX_CHNL_PPS_POS. При выключении приемника по обнулению SYNC_START счетчик сбрасывается. Таким образом, анализируя данный регистр, можно узнать точное положение последнего события PPS относительно начала работы блока по сигналу SYNC_START=1.

Синхронизация применения конфигурации.

Обработка в приемном тракте осуществляется несколькими устройствами последовательно, сигнал передается по цепочке от предыдущего блока к следующему. Многие из блоков обработки имеют память и задерживают сигнал на определенное количество отсчетов. Для того, чтобы конфигурация каждого блока применялась в момент обработки одного и того же отсчета сигнала, предусмотрен механизм синхронизации, принцип действия которого основан на том, что каждый из блоков принимает сигнал применения конфигурации на свой вход SYNC_CFG и передает его через свой выход SYNC_CFG_OUT в следующий в цепочке блок с задержкой, равной задержке сигнала.

Сигнал синхронизации применения конфигурации RX_SYNC_CFG вырабатывается в домене тактовой частоты АЦП и представляет собой импульс длительностью один период тактовой частоты. Он формируется по первому же активному фронту тактовой частоты АЦП, возникающему после изменения состояния поля RX_SYNCCFGOUT регистра SYNC_CTRL с «0» на «1», и подается на вход SYNC_CFG первого в цепочке блока обработки. Сигнал проходит последовательно по цепочке через все блоки обработки, задерживаясь синхронно с обрабатываемыми данными, после чего на тестовом выходе RX_SYNCCFG_OUT формируется импульс длительность которого задается в периодах частоты АЦП через поле SYNCCFGOUT_DLY. Таким образом достигается изменение конфигурации блоков синхронно с прохождением отсчета данных, принятого на входе одновременно с RX_SYNC_CFG.

21.3.1.2 Синхронизация передающих каналов, планировщики задач

Синхронизация передающих трактов может осуществляться как прямым управлением со стороны ПО, так и с использованием автоматизации. Для автоматического синхронного управления передачей в передающем тракте предусмотрены устройства, названные «планировщики задач». Имеется два независимых комплекта планировщиков задач: планировщики применения конфигурации и планировщики старта передачи.

Планировщик применения конфигурации управляет актуализацией конфигураций блоков передающего тракта, осуществляющих обработку сигнала последовательно, один за другим в непрерывном режиме передачи данных. Также, как и в приемном тракте, сигнал SYNC_CFG передается по цепочке блоков обработки вместе с обрабатываемым сигналом, вызывая применение конфигурации блоков обработки синхронно с одним и тем же отсчетом сигнала. Сигнал SYNC_CFG формируется планировщиком в заданный момент времени, привязанный к номеру отсчета на входе передающего тракта.

Планировщик старта передачи задействуется в пакетном режиме передачи данных. Он работает в частотном домене ЦАП и синхронизируется по количеству тактов ЦАП от момента сигнала PPS. При этом поддерживается процедура PUSH – механизм, позволяющий компенсировать задержку, возникающую при обработке сигнала блоками, имеющими внутреннюю память и вносящими задержку в сигнал. Данная процедура заблаговременно заполняет весь передающий тракт данными, вплоть до прихода нужного

отсчета на выход интерфейса с АЦП, и ставит обработку на паузу до получения сигнала синхронизации, который возникает по прошествии заданного количества тактов ЦАП от активного фронта PPS.

21.3.1.2.1 Планировщики применения конфигурации

В блоке DFE_ALT есть 2 планировщика применения конфигурации – по одному для каждого из 2х передающих каналов. Каждый планировщик формирует синхронизирующие сигналы SYNC_CFG для своего канала, которые передаются по цепочке устройств обработки синхронно с прохождением обрабатываемого сигнала. По мере достижения каждого устройства, сигнал приводит к применению в нем нового набора данных конфигурации.

Регистры управления планировщиками конфигурации находятся в пространстве, не адресуемом постранично, поэтому для регистров каждого планировщика выделены отдельные адреса и наименования. Символ x в таблице заменяется на «0» или «1» в соответствии с номером канала, к которому происходит обращение.

Таблица 21.153. Перечень регистров TXSCHED

Название **[значение по сбросу]**	Тип	Бит	Описание
SCHEDx_SYNCCFG_CTRL **[0x0]**	R/W	16	Регистр состояния и управления
SCHEDx_CNT **[0x0]**	R	32	Регистр счетчика отсчетов
SCHEDx_TASK **[0x0]**	R/W	32	Регистр задач

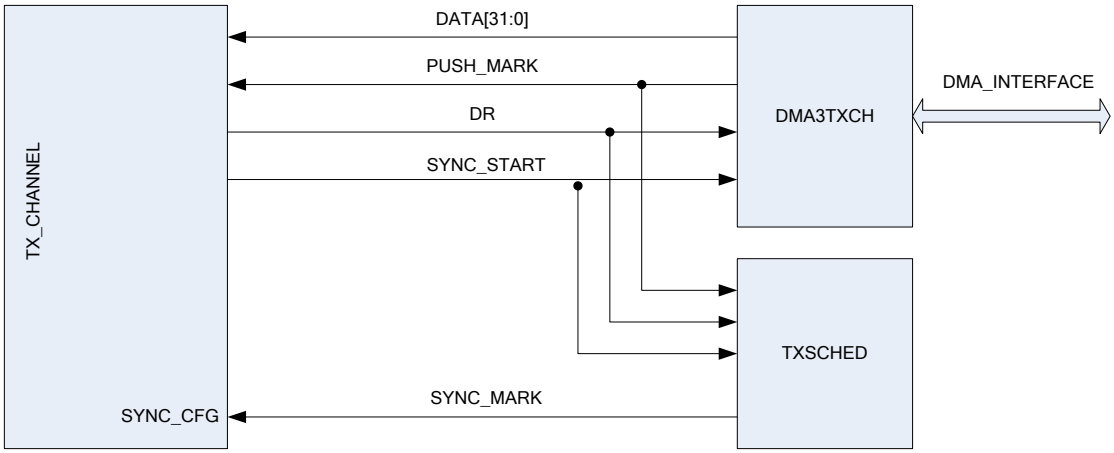


Рисунок 21.10. Блок-схема подключения планировщика

Сигнал SYNC_START используется как сигнал разрешения работы блока. Сигнал работает по уровню, и должен оставаться в состоянии «1» на все время обработки в синхронном режиме. Сигнал PUSH_MARK является маркером валидных данных на выходе DMA. Сигнал DR (Data Request) обозначает готовность передающего канала к принятию данных.

При заполнении выходных FIFO канала передачи данными возможна приостановка чтения данных из FIFO DMA с возобновлением по мере готовности к приему новых отсчетов.

При передаче каждого отсчета сигнала из DMA в передающий тракт происходит инкремент 32 разрядного счетчика отсчетов и его сравнение с заданным числом – номером отсчета, записанным в текущем исполняемом задании синхронизации. При равенстве заданного значения значению счетчика, SYNC_MARK устанавливается в единицу и сбрасывается затем в ноль с событием чтения очередного отсчета. Из-за задержки обработки сигнал SYNC_MARK уходит в передающий тракт синхронно со следующим отсчетом, принятым от DMA. Таким образом, если установлено значение N, то отсчет с номером N+1 в потоке будет промаркирован и станет первым отсчетом, на котором будет применена новая конфигурация.

Счетчик отсчетов работает без обработки события переполнения, т.е. происходит обнуление счетчика при попытке инкремента при его максимальном значении. Обнуление, также, производится при SYNC_START=0, при сбросе по общему сигналу RESET и при записи единицы в поле CNT_RST регистра SCHEDx_SYNCCFG_CTRL.

Блок имеет очередь задач синхронизации глубиной 15, организованную как FIFO. Формирование задания в очереди производится записью номера отсчета в 32 битный регистр SCHEDx_TASK. ПО должно учитывать заполнение очереди при формировании новых задач во избежание их потери. Перед записью нового задания в очередь следует проверять ее заполненность, читая поле SYNC_CFG_LIST регистра SCHEDx_SYNCCFG_CTRL, возвращающее количество заполненных ячеек очереди. При значении поля, равном 15, запись производить нельзя.

Очередь очищается при записи единицы в поле SHED_RST регистра SYNCCFG_CTRL и по системному сбросу.

Импульсы синхронизации формируются только при наличии очередного задания на выходе FIFO при одновременном условии наличия значения «1» в поле SYNC_CFG_ENA регистра SCHEDx_SYNCCFG_CTRL.

Сигнал синхронизации не формируется при значении поля SYNC_CFG_ENA=0. При этом блокируется и вычитывание очереди заданий, но 32-разрядный счетчик отсчетов продолжает работать.

При опустошении FIFO формирование сигнала также блокируется.

Поле CNT_LOCK регистра SCHEDx_SYNCCFG_CTRL предназначено для заморозки возвращаемых значений счетчика отсчетов и текущего задания, что необходимо для синхронизации их чтения. Установка поля в «1» должна предшествовать чтению регистров SCHEDx_CNT и SCHEDx_TASK. В этом случае, содержимое регистров отображает последнее перед заморозкой значение, и чтение их может производиться безопасно в любом

порядке. Чтобы обновить замороженные текущие значения, необходимо установить поле CNT_LOCK в «0», а затем опять в «1».

21.3.1.2.2 Планировщики старта передачи

Основная задача планировщика старта – обеспечить передачу пакета данных строго в определенный момент в локальной шкале времени. Шкала времени задается входными импульсами PPS. Момент начала передачи пакета измеряется в отсчетах частоты ЦАП относительно активного фронта импульса PPS. Получая информацию о позиции строба PPS относительно начала входного потока в приемном тракте из регистров RX_CHNL_PPS_POS, можно выдерживать временные соотношения между принятыми и передаваемыми пакетами с точностью до длительности периода частоты дискретизации.

Следующие регистры используются для управления планировщиком:

Таблица 21.154. Регистры управления планировщиком старта передачи

Название	Тип	Бит	Описание
TX_CHNL_CTRL	R/W	16	Регистр управления/состояния канала
TX_CHNL_TASK_STATE	R/W	16	Регистр управления/состояния планировщика задач
TX_CHNL_TX_DT1 [0xFF]	R/W	24	Регистр установки длительности защитного интервала перед началом передачи пакета
TX_CHNL_TX_DT2 [0xFF]	R/W	24	Регистр установки длительности защитного интервала после окончания передачи пакета
TX_CHNL_TASK_TME [0xFF]	R/W	32	Регистр установки времени старта передачи фрейма. Запись в регистр приводит к добавлению задачи в FIFO

Регистры управления планировщиком находятся в пространстве адресов, доступных через страничный доступ. Текущий режим страничной адресации, задаваемый через регистр TX_CTRL, определяет какой канал или каналы будут конфигурироваться при обращении к регистрам.

Включение в канале режима передачи под управлением планировщика производится установкой в «1» поля MODE регистра TX_CHNL_CTRL.

Планировщик хранит очередь из 16 задач. Постановка задачи в очередь производится путем записи в регистр TX_CHNL_TASK_TME количества отсчетов после события PPS до того момента, на котором необходимо начать передачу, с учетом защитного интервала DT1. При заполнении очереди новые задачи игнорируются. При ее опустошении передача не стартует. ПО может контролироваться глубину заполнения очереди, читая поле TASK_CNT регистра TX_CHNL_TASK_STATE.

При работе под управлением планировщика, перед передачей каждого пакета происходит автоматическая процедура PUSH, обеспечивающая выравнивание начала пакета на выходе интерфейса ЦАП относительно момента времени, заданного в активной (первой в FIFO) задаче. Эта процедура начинается при появлении первых данных нового пакета на выходе интерфейса DMA. Готовность к процедуре PUSH для очередного пакета наступает при

опустошении передающего тракта после передачи предыдущего пакета или при включении канала установкой в «1» поля ENABLE регистра TX_CHNL_CTRL при установленном в «1» поле MODE того же регистра.

ПО должно обеспечивать своевременную готовность данных, передаваемых по каналам DMA, для того чтобы процедуры PUSH успевали завершаться к моменту старта передачи очередного блока данных. В случае срабатывания запроса на старт передачи по активной задаче до окончания процедуры PUSH или до окончания передачи предыдущего пакета, запрос на передачу игнорируется и формируется запрос на прерывание TX_TASK_MISS.

Окончание передачи пакета происходит по сигналу окончания задачи передачи пакета данных от интерфейса DMA. При этом данные с выхода FIFO продолжают читаться в виде нулевых отсчетов, поступая на вход тракта обработки и выталкивая данные на выход вплоть до его опустошения.

В режиме пакетной передачи ПО работает по следующему алгоритму:

1. Разрешение и конфигурирование пакетного режима: TX_CHNL_CTRL.MODE = 1, установка защитных интервалов DT1 и DT2, разрешение работы канала TX_CHNL_CTRL.ENABLE = 1;

2. Формирование первого блока данных для передачи, старт передачи данных по DMA (начало процедуры PUSH). При заполнении тракта обработки данными чтение данных из FIFO DMA приостанавливается до старта передачи. На выходе интерфейса ЦАП все это время формируются нулевые отсчеты.

3. Формирование первой задачи путем записи в TX_CHNL_TASK_TME времени старта с учетом защитного интервала DT1, длительность которого необходимо вычесть из времени начала передачи первого отсчета данных. Передача начнется автоматически, при этом возобновится чтение данных с выхода DMA. При известном расписании передач можно сформировать до 16 задач.

4. Формирование следующего блока данных. Ожидание окончания передачи предыдущего блока, признаком которого является сброс в «0» флага TX_CHNL_CTRL.PM_OUT, что означает окончание процесса выталкивания данных из тракта обработки. При переходе флага из «1» в «0» активируется запрос на прерывание TX_TASK_END;

5. Сброс и запись новой конфигурации устройств тракта обработки (при необходимости), старт новой задачи передачи пакета данных через DMA, при этом стартует новая процедура PUSH;

6. Пополнение (при необходимости) FIFO задач и переход к пункту 4.

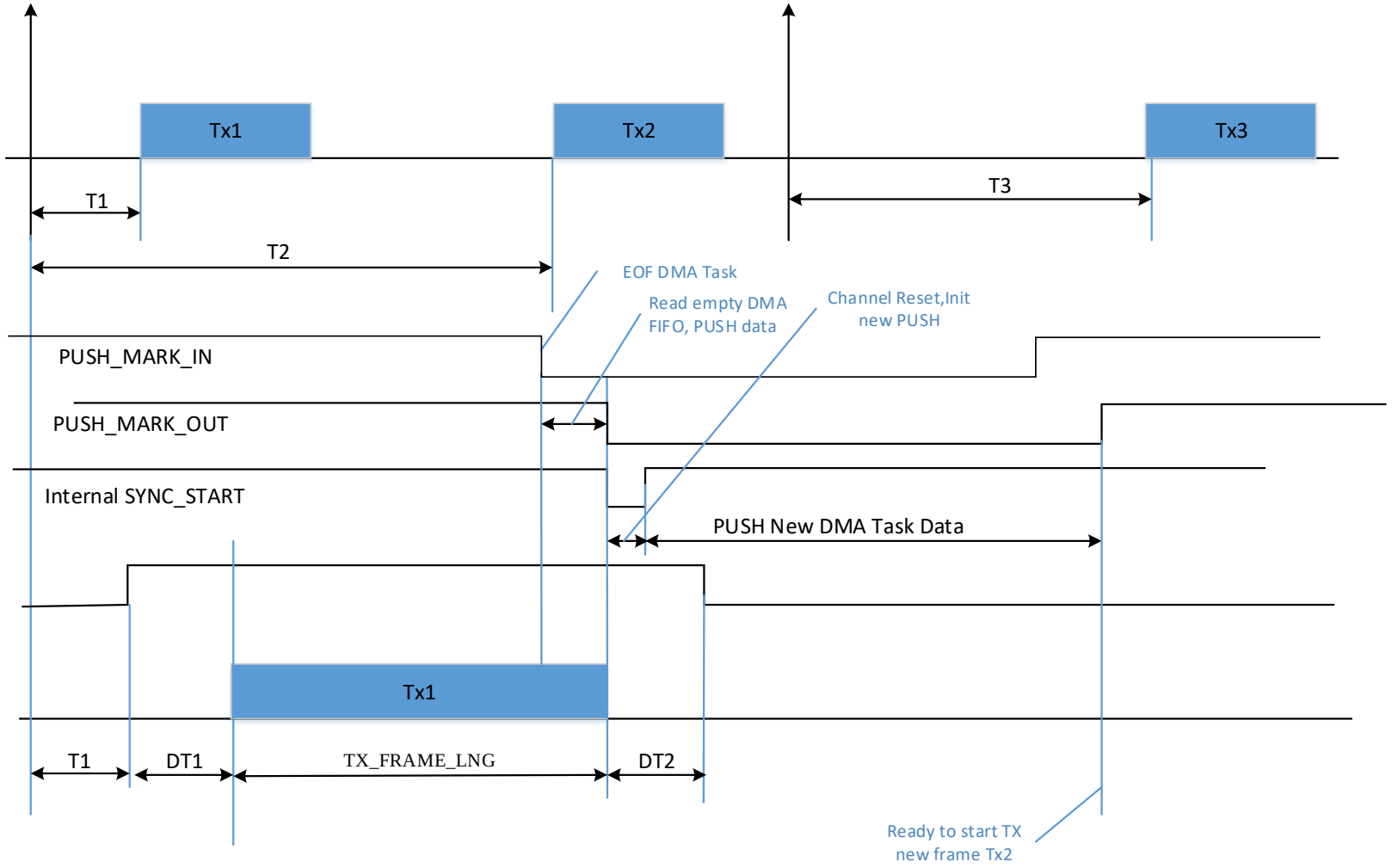


Рисунок 21.11. Фазы пакетной передачи в TX_CHNL

На Рисунок 21.11 описаны фазы передачи пакетов.

Время старта передачи пакетов T1, T2 и T3 определяется временем, заданным в задачах, загруженных в FIFO через регистр TX_CHNL_TASK_TME. Стоит обратить внимание, что в примере на рисунке время T3 отсчитывается от следующего события PPS, произошедшего после старта T1 и T2. Это необходимо учитывать при формировании очереди задач.

В начале передачи устанавливается сигнал TX_MARK, разрешающий работу внешних устройств. Если предыдущий фрейм еще не снял TX_MARK, то он остается активным до завершения текущей операции передачи. На время защитного интервала DT1, предназначенного для компенсации времени переходных процессов во внешних устройствах, на выход интерфейса ЦАП продолжают выдаваться нули. По окончании DT1 (задается в регистре TX_CHNL_DT1) разрешается выдача отсчетов и продвижение данных по каналу обработки. Передача продолжается пока на выходе канала не упадет флаг PUSH_MARK_OUT. Это соответствует случаю, когда все данные были вычитаны по DMA и продвинуты по всему каналу до самого выхода. По окончании передачи вырабатывается сигнал прерывания TX_TASK_END и происходит сброс канала обработки. После этого инициируется процедура PUSH для следующего пакета. Сброс сигнала TX_MARK после окончания передачи пакета задерживается на длительность защитного интервала DT2 (задается в регистре TX_CHNL_DT2).

В случае, если для следующего пакета требуется иная конфигурация блоков обработки, то на время ее задания канал должен быть выключен полем ENABLE регистра TX_CHNL_CTRL.

Если FIFO задач пусто, и на его выходе задача отсутствует, формирования сигналов на передачу фреймов не происходит.

Если на выходе FIFO присутствует задача, значение ее счетчика всегда выставлено на вход блока сравнения и запуска. При активизации передачи, она удаляется с выхода FIFO и немедленно активизируется следующая задача из очереди. Это необходимо учитывать для того, чтобы не допускать перекрытия передач пакетов и опозданий завершения процедур PUSH. Для отладочных целей предусмотрен механизм детектирования подобных ситуаций: при их возникновении вырабатывается запрос на прерывание TX_TASK_MISS, новая задача отменяется, работа планировщика останавливается (SCH_ENABLE=0). Интервал чувствительности TX_TASK_MISS начинается с момента установки TX_MARK и заканчивается моментом окончания процедуры PUSH следующего блока – новая задача считается пропущенной если блок данных для нее не успел пройти процедуру PUSH к моменту старта. TX_TASK_MISS активен только в режиме планировщика.

Регистр TX_CHNL_TASK_TME при записи в него добавляет значение в очередь задач, при чтении возвращает значение времени старта для первой в очереди задачи. В случае TX_TASK_MISS возвращаемое значение соответствует задаче, которая привела к пропуску передачи фрейма.

Поле PPS_EVENT регистра TX_CHNL_CTRL устанавливается в «1» по событию PPS и сбрасывается в «0» при выключении канала прямой записью в него.

21.3.1.2.3 Прямая синхронизация старта

При работе без использования планировщика задач сигналы синхронизации старта передающего тракта TX_SYNC_START формируются аналогично сигналам приемника – по фронту сигнала PPS или по факту записи «1» в поле TX_SYNC_START.

События PPS считаются 4-разрядным счетчиком, отображаемым на поле PPS_CNT регистра TX_CHNL_TASK_STATE. Счетчик PPS сбрасывается общим Reset, и SYNC_START и полем ENABLE канала.

Сигнал разрешения работы внешних устройств TX_MARK устанавливается в единицу если канал включен и входной SYNC_START равен единице в режиме без планировщика.

21.3.1.3 Точная подстройка задержки тактового сигнала АЦП и ЦАП

DFE обладает возможностью подстройки фазы тактовых сигналов дискретизации АЦП и ЦАП путем регулировки их задержки и инвертирования. Целью регулировки задержки тактового сигнала является компенсация задержки выставления данных на выходе АЦП и задержек распространения сигнала по линиям печатной платы и соединениям внутри корпуса ИМС.

Каждый из входных тактовых сигналов портов АЦП и ЦАП проходит через блок регулировки задержки прежде, чем он поступит на глобальную тактовую сеть и будет применен в АЦП и ЦАП интерфейсах. Шаг подстройки задержки составляет ~100pS, количество возможных вариантов задержки – 16.

Величина задержки тактового сигнала АЦП задается через поле ADC_CLKDLY регистра CLK_SYNC_CFG, а его инверсия включается установкой поля ADC_CLKINV в состояние «1». Для тактового сигнала ЦАП необходимо использовать поля DAC_CLKDLY и DAC_CLKINV того же регистра.

Блок схема регулировки фазы на примере тактового сигнала ЦАП представлена на Рисунок 21.12.

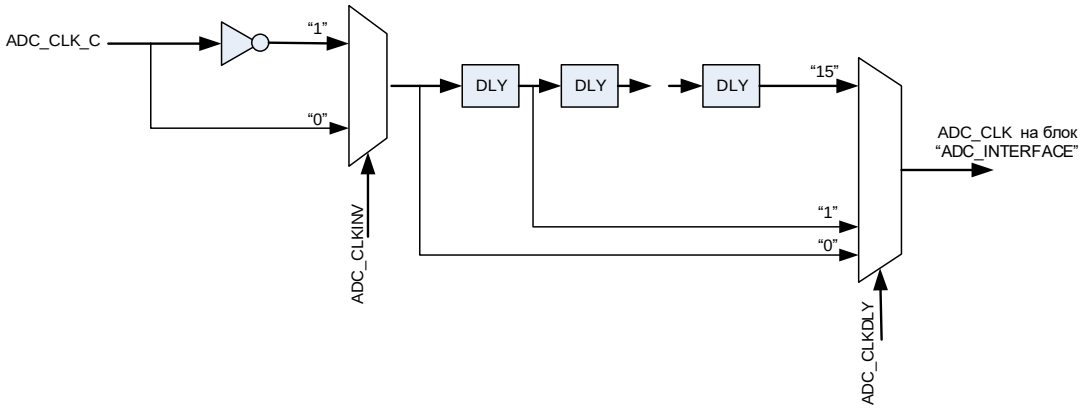


Рисунок 21.12. Блок-схема управления фазой тактовых сигналов

21.3.1.4 Управление DMA каналами RX

Каналы DMA управляются через регистры DMARX12_CTRL, DMARX34_CTRL, DMARX56_CTRL. Каждый регистр управляет парой каналов, номера которых отражены в его наименовании. Все поля управления, перечисленные ниже относятся к данным регистрам, применительно к каналу с нужным номером.

Выходные шины данных вычислительных узлов RX части подключаются к входной шине данных DMA в соответствии с таблицей, приведенной ниже.

Таблица 21.155. Подключение источников сигналов к RX DMA

Источник входных данных DMA	Разряды входной шины данных DMA интерфейса DMA_IN			
	63..38	37..32	31..6	5:0
CH1…CH4	CHx_I[25:0]	0	CHx_Q[25:0]	0
SUM	SUM_I[25:0]	0	SUM_Q[25:0]	0
MTFLT	MTFLT[63:0]			
PROBE	PROBE_DATA_OUT[31:0]		0	

Выходные отсчеты обеих компонент выходных сигналов приемных каналов и сумматора, имеющие разрядность 26 бит, дополняются нулями со стороны младших разрядов до разрядности 32 бита, после чего Q-компонента помещается в младшую половину 64-битного слова входных данных DMA контроллера DMA_IN, а I-компонента – в старшую.

Отсчеты сигналов, принятые на 64-разрядный вход контроллера DMA, могут передаваться как в неизмененном виде, так и с упаковкой по несколько отсчетов в одном 64-разрядном слове, передаваемом по шине AXI. При необходимости, разрядность данных усекается путем отбрасывания младших битов. Режим упаковки управляется полем DMARXx_PACK_MODE (сокр. PACK_MODE). Все режимы упаковки отсчетов в 64-разрядное слово представлены в таблице.

Таблица 21.156. Режимы упаковки отсчетов RX DMA

PACK_ MODE	63:56	55:48	47:40	39:32	31:24	23:16	15:8	7:0
0	DMA_IN[63:0]							
1	DMA_IN1[63:32]				DMA_IN0[63:32]			
2	DMA_IN1[63:48]		DMA_IN1[31:16]		DMA_IN0[63:48]		DMA_IN0[31:16]	
3	DMA_IN3 [63:56]	DMA_IN3 [31:24]	DMA_IN2 [63:56]	DMA_IN2 [31:24]	DMA_IN1 [63:56]	DMA_IN1 [31:24]	DMA_IN0 [63:56]	DMA_IN0 [31:24]

При установке PACK_MODE=0 принятое 64-бит слово записывается в FIFO без изменений. Этот режим используется в следующих случаях:

- вывод выходных сигналов RX каналов или сумматора с упаковкой 1 отсчет в слове без потери разрядности отсчетов;
- вывод результатов работы согласованного фильтра (4x16 бит) с упаковкой 1 отсчет в слове.

При PACK_MODE=1 из каждого входного слова берутся старшие 32 бита и последовательно упаковываются в 64 разрядное слово начиная с его младших разрядов (первый отсчет записывается в младшие 32 бита слова, второй – в старшие). Этот режим используется в случаях:

- вывод 32-разрядного тестового сигнала Rx тракта с упаковкой 2 отсчета в слове.
- вывод 32-разрядного тестового сигнала Tx тракта с упаковкой 2 отсчета в слове.
- вывод 26-разрядной I компоненты выходного сигнала Rx канала или сумматора с упаковкой 2 отсчета на слово без потери разрядности.

При PACK_MODE == «2» или «3» при упаковке из старшей и младшей половин каждого входного слова берутся старшие 16 и 8 бит, а младшие биты игнорируются. Таким образом достигается усечение разрядности I и Q компонент отсчетов соответственно. В режиме «2» передаются по 2 отсчета на слово, в режиме «3» – по 4 отсчета. Эти режимы используются для снижения нагрузки на шины AXI ценой потери динамического диапазона выходных сигналов.

21.3.1.5 Управление DMA каналами TX

В передающем тракте выходы каналов DMA подключены непосредственно ко входам каналов TX. Порт DMA записывает в FIFO данные в «сыром» формате. Это означает, что 32 разрядные данные будут интерпретироваться в блоке обработки TX тракта в зависимости от режима работы модулятора.

Принцип упаковки отсчетов в 64-бит слово представлен в таблице ниже.

Таблица 21.157. Упаковка отсчетов DMA2TXCH

63:32	31:0
DATA_IN1[31:0]	DATA_IN0[31:0]

Это 32 разрядное слово может впоследствии быть интерпретировано как 16 бит I и Q отсчеты сигнала, либо как упакованная символьная информация (см. описание передающего тракта).

21.3.1.6 Контроллер прерываний

В состав DFE_ALT входит собственный блок контроллера прерываний. Управление им осуществляется с помощью четырех групп регистров – запроса, маскирования, сброса и выбора режима по фронту или уровню. Прерывание от каждого источника управляется битом, находящимся в одной и той же позиции в каждой группе регистров.

Запись «1» в определенный разряд регистра маскирования x_MASK разрешает формирование сигнала прерывания по возникновению состояния «1» в соответствующем разряде регистра запроса, но не влияет на состояние последнего.

Биты регистров запроса устанавливаются в состояние «1» по выполнению условия в соответствующем источнике независимо от значения регистров маскирования.

Сброс прерывания осуществляется записью единицы в соответствующий бит регистра сброса прерывания x_RESET. Сброс имеет низший приоритет перед установкой прерывания при их одновременном возникновении.

Каждое прерывание формируется в поле регистра запроса либо по уровню, либо по фронту сигнала источника, что определяется полями регистров выбора режима x_SENSE. Значения «1» в полях регистров x_SENSE соответствуют установке режима по фронту. В этом случае в поле регистра запроса записывается «1» по событию перехода входного сигнала прерывания из нулевого в единичное состояние. В случае установки режима «по уровню», значение поля повторяет значение сигнала прерывания.

Контроллер имеет двухуровневую структуру приоритетов прерываний. Регистр запросов на прерывания IRQ и соответствующие ему регистры IRQ_MASK, IRQ_RESET и IRQ_SENSE обслуживают прерывания первого уровня, требующие наиболее быстрой реакции.

Прерывания второго уровня не требуют оперативного вмешательства, т.к. либо это прерывание, сигнализирующее о фатальной ошибке, и скорость реакции внешних устройств уже не играет роли в исправлении ошибки, либо это прерывание может быть отложено и обработано позднее. Например, фатальная ошибка по переполнению FIFO приводит к серьезному сбою, и скорость реакции не сможет уже исправить ситуацию, а при возникновении перегрузки в канале не требуется оперативного вмешательства.

Прерывания второго уровня сгруппированы в 4 группы, каждой из которой соответствует одно прерывание первого уровня и один бит в регистрах управления IRQ_x:

- бит DMA_INT – для встроенного контроллера DMA;
- бит OVF_INT – для контроля перегрузки в каналах обработки;
- бит SE_RDY – готовность оценок параметров сигналов в коммутаторах приемного и передающего трактов;
- бит DRE_EST_RDY – готовность оценок параметров сигналов в блоках расширения динамического диапазона DRE.

Группа прерываний от встроенного контроллера DMA управляется регистрами DMA_INT, DMA_INT_MASK, DMA_INT_RESET, DMA_INT_SENSE.

Для остальных трех групп выделены диапазоны битов в регистрах LP_INT, LP_INT_MASK, LP_INT_RESET, LP_INT_SENSE.

Каждое прерывание второго уровня сначала обрабатывается с учетом собственных значений битов запроса, маскирования, режима и сброса. Выходные сигналы всех прерываний каждой группы объединяются логической операцией «или» в выходной сигнал группы и подаются на соответствующий бит регистра запросов прерываний первого уровня. Далее, групповые сигналы обрабатываются как сигналы первого уровня.

В ПО во время обработки прерывания, при обнаружении установленного бита группового прерывания, необходимо выяснить конкретную причину прерывания путем анализа регистров запросов DMA_INT и LP_INT. После обработки необходимо осуществить сброс прерывания сначала внутри группы, пользуясь регистрами DMA_INT_RESET и LP_INT_RESET, затем сбросить прерывание от группы через регистр IRQ_RESET. При нарушении порядка сброса возможно формирование ложного повторного прерывания.

Рекомендованным режимом работы контроллера является режим по фронту, т.к. он позволяет избежать возникновения повторных прерываний после сброса при не устраненной причине. Например, сигналы прерываний типа DMARX1_FULL и DMATX1_EMP являются флагами состояния FIFO, которые меняют свое состояние при обращениях к FIFO, и в случае работы по уровню, после сброса прерывания, если флаг еще активен, запрос на прерывание сформируется вновь на следующем такте после сброса.

В случае работы по уровню, необходимо сначала устранить причину (например, вычитать FIFO), а затем сбросить прерывание обычным способом через регистр.

21.3.1.7 Блок записи выборки сигнала PROBE

Данный блок предназначен для отладочных целей. Он позволяет записать выборку длиной до 2048 отсчетов из высокоскоростных потоков данных, подающихся на тестовые выходы

блоков коммутации приемного или передающего трактов, в собственный блок памяти размером 2048 32-разрядных слов. После остановки записи данные могут быть прочитаны либо процессором через регистр PROBE_DATA, либо контроллером DMA. Тестовые выходы приемного и передающего коммутаторов активны только при RX_SYNCSTART=1.

Блок управляется через регистр PROBE_CTRL. Через него выбирается один из двух источников сигнала (поле SRC), размер блока (поле LNG_REQ), способ считывания памяти (поле DST) и осуществляется старт записи путем установки поля START в «1». Поля STATE, IS_EMPTY и FULLNESS позволяют контролировать процесс работы.

Размер записываемого блока, определяемый полем LNG_REQ, может быть выбран из набора 512, 1024 или 2048 отсчетов заданием значений 1, 2 или 3 соответственно. При LNG_REQ = 0 работа блокируется, запись не производится.

При старте процесса записи поля STATE и FULLNESS принимают значение «0». Поле IS_EMPTY устанавливается в «1», сигнализируя о пустоте буфера и сбрасывается в «0» при записи в буфер первого отсчета. Степень наполненности блока данных в процессе записи отражается в поле FULLNESS. Его значения соответствуют тем же порогам, что и значения поля LNG_REQ, т.е. 0, 512, 1024 и 2048 отсчетов.

Остановка записи происходит по достижению заполнения заданного объема, т.е. при FULLNESS = LNG_REQ. При этом STATE устанавливается в «1» и вырабатывается сигнал прерывания.

Если в процессе записи данных поступает команда на старт записи, то текущая процедура прерывается и начинается сначала.

После останова записи можно начинать процедуру считывания данных. Данные, прочитанные до окончания записи, не являются валидными. Способ считывания определяется через поле DST. При DST=1 считывание записанного блока осуществляется через регистр PROBE_DATA, при DST=0 – по внутренней скоростной шине при помощи контроллера DMA. Чтение осуществляется последовательно, в том же порядке, в каком данные поступали на вход. При вычитывании последнего слова поле IS_EMPTY устанавливается в единицу. Поле FULLNESS при чтении данных не изменяется.

Источник данных для записи выбирается полем SRC регистра PROBE_CTRL которое выбирает между приемным и передающим трактами.

При выборе приемного тракта селекция источника сигнала осуществляется в коммутаторе приемного тракта полем PROBE_SRC регистра RX_IN_DTRT_TST_CTRL и описана в таблице.

Таблица 21.158. Коммутация потоков входных данных для блока записи выборки сигнала

PROBE_SRC	TEST_DATA_OUT

0…3	В режиме OCHx_CPLX=0: компонента I выхода коммутатора 0 – 3, расширенная знаком до 32 бит. TEST_DATA_OUT[31:0] = {SSSSSS CHx_I_OUT[25:0]}, где S = CHx_I_OUT[25]. В режиме OCHx_CPLX=1: усеченные до 16 разрядов I и Q компоненты выхода коммутатора 0 – 3, Q-компонента в младшей части 32-битного слова. TEST_DATA_OUT[31:16] = CHx_I_OUT[25:10] TEST_DATA_OUT[15:0] = CHx_Q_OUT[25:10]
4	Данные с АЦП (входы коммутатора) TEST_DATA_OUT[31:16] = CH1_IN[15:0] – канал 1 TEST_DATA_OUT[15:0] = CH2_IN[15:0] – канал 2
5	Данные с АЦП (входы коммутатора) TEST_DATA_OUT[31:16] = CH3_IN[15:0] – канал 3 TEST_DATA_OUT[15:0] = CH4_IN[15:0] – канал 4
6	Сигналы после коррекции смещения нуля в коммутаторе TEST_DATA_OUT[31:16] = CH1_17[16:1] – канал 1 TEST_DATA_OUT[15:0] = CH2_17[16:1] – канал 2
7	Сигналы после коррекции смещения нуля в коммутаторе TEST_DATA_OUT[31:16] = CH3_17[16:1] – канал 3 TEST_DATA_OUT[15:0] = CH4_17[16:1] – канал 4

Подача данных на выход TEST_DATA_OUT должна быть разблокирована установкой поля PROBE_ENA регистра RX_IN_DTRT_TST_CTRL в «1».

При выборе для записи в буфер PROBE сигналов передающего тракта селекция источника сигнала осуществляется полем PROBE_SRC регистра TX_OUT_DTRT_TST_CTRL в коммутаторе передающего тракта. Выбор производится из 2 вариантов по таблице:

PROBE_SRC	TEST_DATA_OUT
0	Данные, подающиеся на ЦАП (выходы коммутатора) TEST_DATA_OUT[31:16] = DAC_CH2[15:0] – канал 2 TEST_DATA_OUT[15:0] = DAC_CH1[15:0] – канал 1
1	Данные, подающиеся на ЦАП (выходы коммутатора) TEST_DATA_OUT[31:16] = DAC_CH4[15:0] – канал 4 TEST_DATA_OUT[15:0] = DAC_CH3[15:0] – канал 3

Подача данных на выход TEST_DATA_OUT должна быть разблокирована установкой поля PROBE_ENA регистра TX_OUT_DTRT_TST_CTRL в «1».

Примечание: при работе в двухканальных режимах старшая и младшая половины 32-битного слова коммутируются по-разному в приемном и передающем тракте. В приемном тракте в старшей половине расположен отсчет канал с меньшим номером, в передающем – наоборот, с большим.

21.3.1.8 Приемная часть DFE

Функциональная схема приемной части приведена на рисунке ниже.

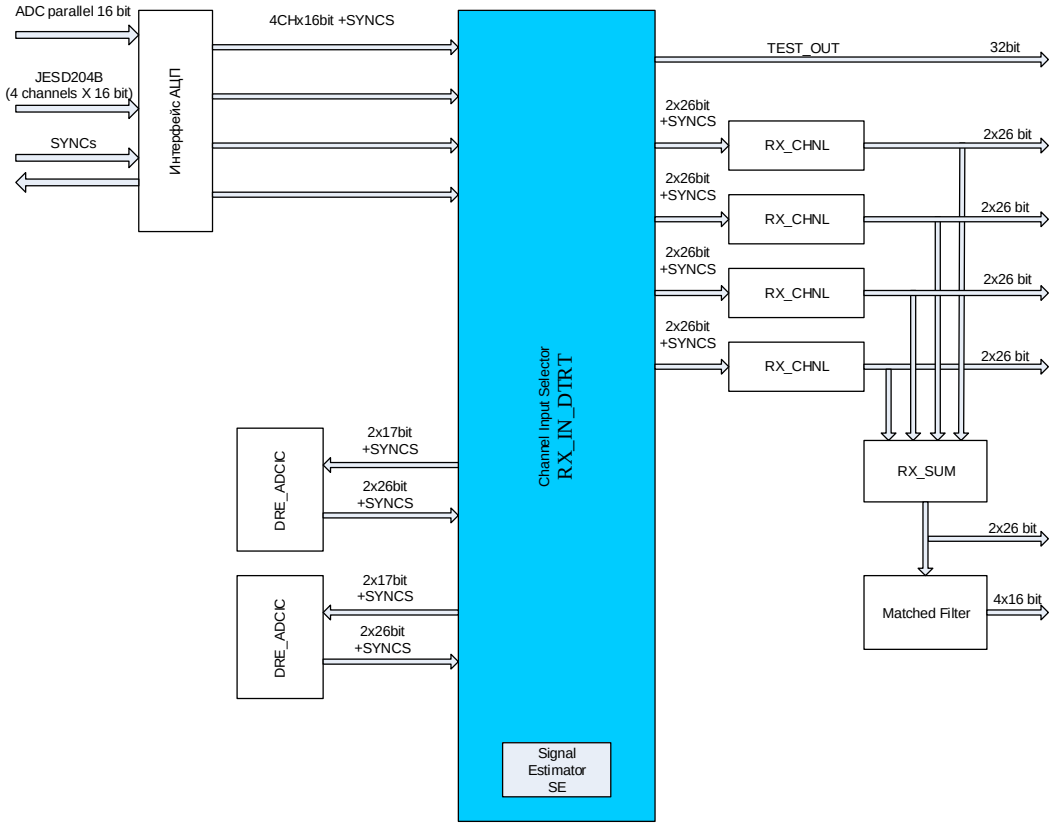


Рисунок 21.13. Блок-схема приемной части DFE

Приемная часть состоит из следующих модулей.

- Интерфейс с АЦП – ADCINTERFACE. Блок предназначен для приема данных от АЦП в домене тактового сигнала АЦП в режимах SDR или DDR, приема до четырех потоков данных по интерфейсу JESD204B и формирования потока данных в тактовом домене обработки CLK.

- Блок расширения динамического диапазона АЦП и компенсации небаланса каналов – DREADCIC. В зависимости от режима воспринимает свои входные потоки либо как оцифрованный двумя разными АЦП с различными коэффициентами усиления действительный сигнал, либо как один комплексный сигнал. В первом случае (режим DRE) блок формирует один действительный сигнал с большим динамическим диапазоном. Во втором случае формирует один комплексный сигнал с компенсированным небалансом амплитуд и фаз.

- Селектор источников входных сигналов – RX_IN_DTRT. Селектор предназначен для выбора источника сигнала для блоков обработки приемной части. Блок осуществляет выбор источника и приведения его в зависимости от контекста (входной сигнал действительный или аналитический) к формату входных данных блоков обработки. Дополнительно блок выполняет коррекцию смещения нуля сигналов АЦП и оценку их параметров.

- Приемные каналы цифровой обработки сигнала – RX_CHNL. 4 канала осуществляют перенос частоты, децимацию и фильтрацию сигнала.
- Сумматор каналов - RX_SUM выполняет весовое суммирование двух RX_CHAN, работающих в синхронном режиме.
- Согласованный фильтр (комплексный FIR 128 порядка) – MTCFLT128. Осуществляет свертку комплексного сигнала с импульсной характеристикой длиной до 128 комплексных восьмибитных отсчетов. Для свертки с максимальной длиной импульсной характеристики требуется 32 такта на один входной отсчет.

21.3.1.8.1 Интерфейс с АЦП – ADCINTERFACE

Блок предназначен для приема потоков данных от АЦП по 16 бит параллельным шинам или JESD204B интерфейсу, синхронизации с внутренней тактовой частотой и формирования входных данных для блоков ЦОС приемной части DDC. Поддерживаются следующие варианты интерфейсов.

- Простой параллельный интерфейс
  - SDR
  - DDR с мультиплексированием данных
  - DDR с мультиплексированием каналов
  - DDR с мультиплексированием (уплотнением) отсчетов
- JESD204B
- JESD207
  - SDR
  - DDR
- Оригинальный JESD207 подобный интерфейс трансиверов AD9361

При работе по параллельным интерфейсам в режимах SDR и DDR поддерживается прямо-смещенный, двоично-дополнительный код, код Грэя и скремблирование данных. В режиме совместимости с JESD207 – только двоично-дополнительный код.

При разрядности АЦП меньшей, чем разрядность порта, его подключение должно осуществляться к старшим разрядам портов.

Выбор режима работы интерфейса осуществляется через поля регистра ADC_CTRL.

Поле SRC_SEL выбирает порт (CMOS, LVDS, JESD204B) или выключает интерфейс при записи «0». Порты CMOS и LVDS обеспечивают прием потоковых данных в SDR и DDR режимах и могут работать в режиме совместимости с SDR/DDR режимами JESD207.

Поле PBUS_MODE задает режим работы (DDR/SDR/JESD207), и режим мультиплексирования DDR параллельных портов, а поле MODE_OPT1 определяет количество каналов и их разрядность. В зависимости от состояния поля PBUS_MODE, значение поля MODE_OPT1 приводит к установке разных конфигураций. Полный набор режимов представлен в таблице.

Таблица 21.159. Режимы работы ADCINTERFACE

	MODE_OPT1		Data Rate	Frame Sync
Simple Parallel SDR PBUS_MODE=0	0	1Ch X 16bit	ADC CLK	-
	1	2Ch X 8bit	ADC CLK	-
Simple Parallel DDR мультиплексирование каналов PBUS_MODE=1	0	2Ch X 16bit	ADC CLK	-
	1	4Ch X 8bit	ADC CLK	-
Simple Parallel DDR мультиплексирование бит PBUS_MODE=2	0,1	2Ch X 16bit	ADC CLK	-
JESD207 SDR (AD9361 CMOS) PBUS_MODE=4	0	2Ch X 15bit	ADC CLK/2	Есть (LSB)
	1	4Ch X 15bit	ADC CLK/4	Есть (LSB)
JESD207 DDR (AD9361 CMOS) PBUS_MODE=5	0	2Ch X 15bit	ADC CLK	Есть (LSB)
	1	4Ch X 15bit	ADC CLK/2	Есть (LSB)
AD9361 DDR LVDS (DDR) PBUS_MODE=6	0	2Ch X 15bit	ADC CLK	Есть (LSB)
	1	4Ch X 15bit	ADC CLK/2	Есть (LSB)

Поле MODE_OPT2 используется только в режиме совместимости JESD207 и описано в соответствующем разделе.

В режимах JESD204 и JESD207 все поля, не относящиеся к их конфигурации, должны быть установлены в значения «по сбросу».

На выходе интерфейса АЦП формируются четыре 16 разрядных потока данных, синхронных между собой. В поле OUT_DATA_BW регистра DATA_FMT_AUX необходимо задать актуальную разрядность данных АЦП. В случае, если разрядность АЦП меньше разрядности портов, неиспользуемые младшие биты будут обнулены.

Блок-схема в части формирования потоков представлена на Рисунок 21.14.

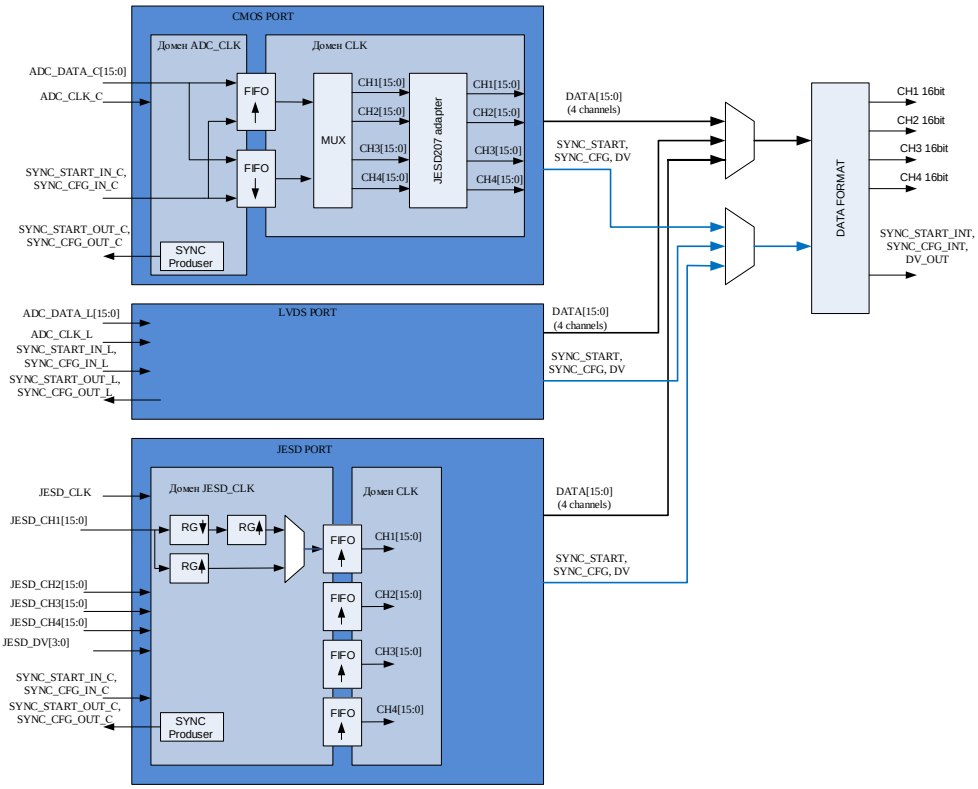


Рисунок 21.14. Блок-схема обработки в ADCINTERFACE

FIFO предназначены для буферизации сигнала при переходе между тактовым доменом АЦП и внутренним системным доменом обработки.

Блок DATAFORMAT осуществляет преобразование формата данных и декодирование. Результатом работы блока являются отсчеты сигналов в двоично-дополнительной кодировке нужной разрядности. Блок может осуществлять дескремблирование, преобразование из кодов Грэя и прямо-смещенного кода в двоично-дополнительный. Данные операции совместимы только с режимами PBUS_MODE=0,1,2 при выборе параллельных портов CMOS или LVDS (SRC_SEL=1,2). В режимах JESD207-совместимом и JESD204B данные всегда передаются непосредственно в двоично-дополнительном коде без скремблирования, и устройства должны быть отключены.

В каждом из 4 каналов данных блок содержит цепочку из 4 устройств, представленных на рисунке.

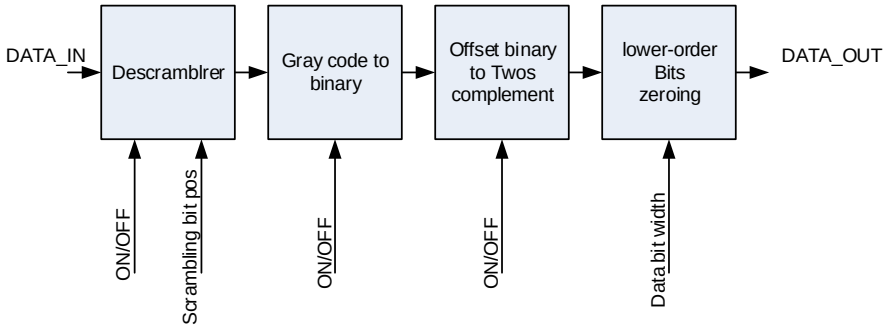


Рисунок 21.15. Блок-схема преобразование формата данных АЦП

Первый блок – дескремблер – включается установкой в «1» поля DSCRMB_ON регистра ADC_CTRL. Блок необходим для обеспечения совместимости с рядом моделей АЦП от фирмы Linear Technology в которых скремблирование применяется с целью обелить спектр наводок с шины данных АЦП на его вход и тем самым снизить уровень узкополосных помех. Поле DSCRMB_BPOS регистра DATA_FMT_AUX задает параметр, необходимый для работы устройства – номер ведущего бита.

Дескремблирование для всех битов, кроме ведущего выполняется операцией XOR с ведущим битом, как представлено в примере на Рисунок 21.16. В данном случае, в качестве ведущего выбран бит 2.

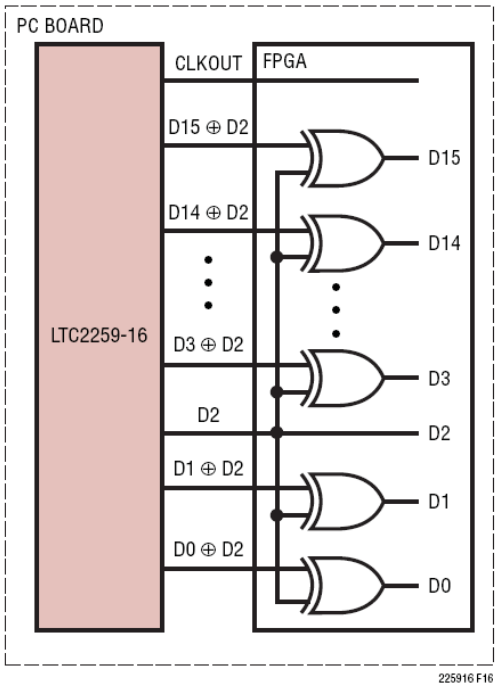


Рисунок 21.16. Пример дескремблирования шины данных АЦП

Пример дескремблера выполняет операцию, обратную по отношению к операции скремблирования с битом D2, производимой в ИМС АЦП, приведенной на рисунке ниже.

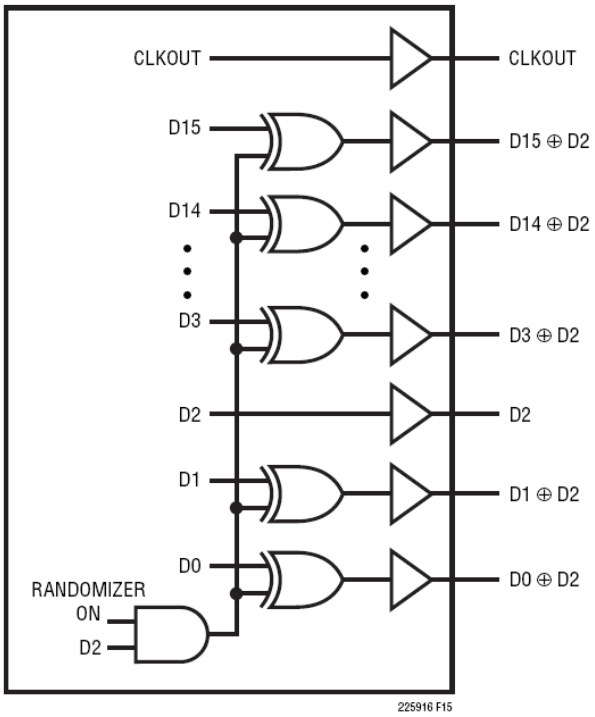


Рисунок 21.17. Пример скремблирования шины данных АЦП

Любой бит из 16 может быть задан в качестве ведущего, но при этом необходимо следить за тем, чтобы позиции бита в обоих алгоритмах совпадали, учитывая, что АЦП меньшей разрядности подключается к старшим битам шины.

Вторым блоком в цепочке является преобразователь кода грэя в двоичный код. Он включается установкой поля GRAYDECODE_ON регистра ADC_CTRL в состояние «1». Преобразование выполняется по известному алгоритму, приведенному на Рисунок 21.18 где b(1) – старший бит (MSB).

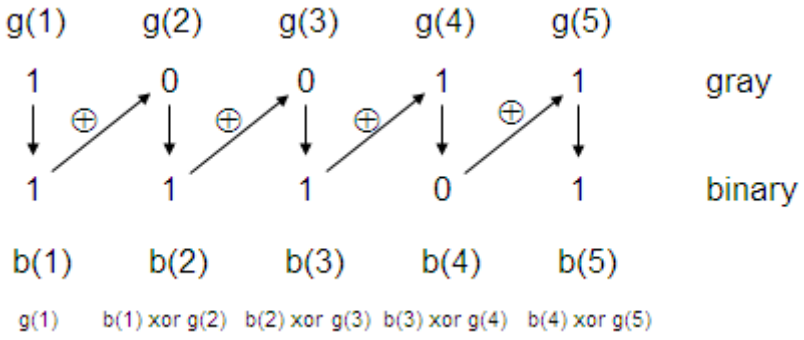


Рисунок 21.18. GRAY to BIN декодер шины данных АЦП

Третий блок производит преобразование прямо-смещенного кода в двоично-дополнительный и включается установкой в состояние «1» бита OB2TC_ON регистра ADC_CTRL. Преобразование осуществляется инверсией старшего бита, но прибавление

единицы не производится, поскольку в блоке коммутации имеется возможность коррекции смещения нуля прибавлением любого числа.

Поскольку дескремблирование и декодирование кода Грэя влияет на все разряды, в случае применения АЦП с меньшей чем 16 разрядов шиной данных, необходимо дополнительно обнулить незадействованные биты. АЦП подключается с выравниванием по старшим разрядам, поэтому последний блок в цепочке обнуляет младшие разряды. Количество значащих разрядов на выходе устройства задается полем OUT_DATA_BW регистра DATA_FMT_AUX. Число, записываемое в поле, является уменьшенной на 1 требуемой разрядностью. Данный блок совместим с любым режимом работы интерфейса АЦП.

21.3.1.8.1.1 Формат принимаемых данных в CMOS и LVDS портах АЦП в режиме простой SDR параллельной шины данных.

Данные передаются по возрастающему фронту тактового сигнала АЦП с темпом один отсчет на период. Данные могут интерпретироваться интерфейсом как

- один канал с разрядностью до 16 бит;
- два канала с разрядностью до 8 бит.

Таблица 21.160. Упаковка данных в SDR параллельной шине данных АЦП

	SDR 16bit MODE_OPT1=0	SDR 8bit MODE_OPT1=1
CH1[15:0]	ADC_DATA[15:0]	ADC_DATA [15:8] выравнивание по старшим битам, младшие биты заполняются нулем
CH2[15:0]	0	ADC_DATA [7:0] выравнивание по старшим битам, младшие биты заполняются нулем
CH3[15:0]	0	0
CH4[15:0]	0	0

21.3.1.8.1.2 Формат принимаемых данных в CMOS и LVDS портах АЦП в режиме простой DDR параллельной шины данных.

Данные передаются по обоим фронтам тактового сигнала с темпом один отсчет на период тактового сигнала. При этом существует следующие варианты передачи данных:

- На шине мультиплексируются (чередуются) биты отсчетов АЦП;
- На шине мультиплексируются сами отсчеты АЦП.

Режим DDR с мультиплексированием бит всегда интерпретируется как «два канала по 16 бит». При этом 16 разрядная шина данных АЦП интерпретируется как две шины по 8 бит. По старшим 8 битам шины передаются данные первого канала в режиме DDR, по младшим 8 бит – DDR данные второго канала.

При мультиплексировании бит на шине существуют два варианта мультиплексирования:

- в отсчете первым передается младший бит из мультиплексируемой пары разрядов;

- в отсчете первым передается старший бит.

На рисунке ниже приведен пример мультиплексирования битов данных, когда в отсчете передается сначала младший бит. Мультиплексирование бит осуществляется в паре четный/нечетный бит: 15/14, 13/12, …. 1/0.

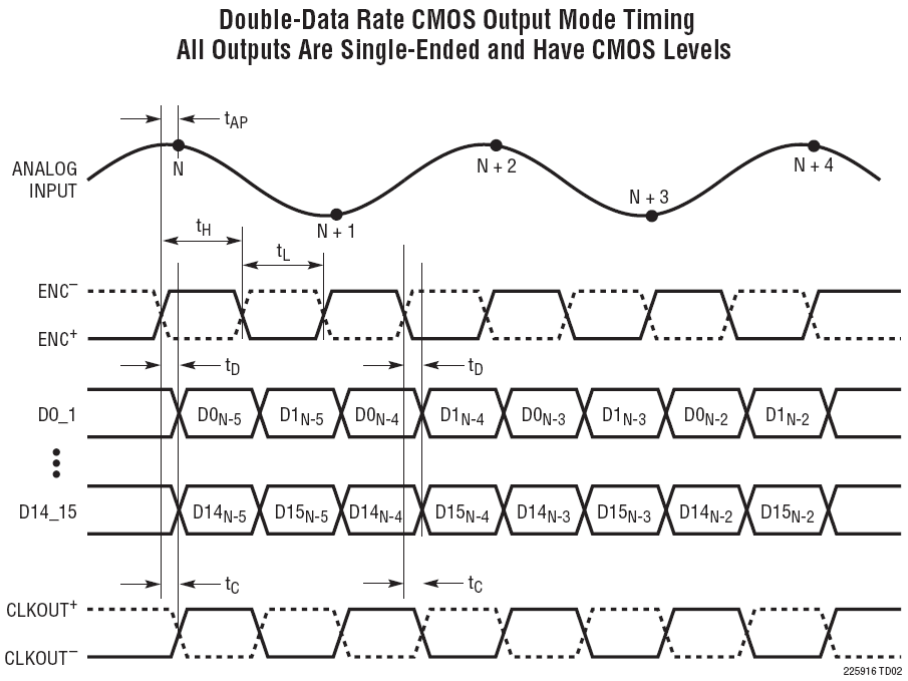


Рисунок 21.19. Временная диаграмма примера мультиплексирования бит в DDR режиме интерфейса АЦП

Данные в режиме DDR с мультиплексированием каналов могут интерпретироваться двумя способами.

- Два канала с разрядностью до 16 бит.
- Четыре канала с разрядностью до 8 бит. Биты 15:8 соответствуют паре каналов CH1 и CH2, биты 7:0 соответствуют другой паре каналов CH3 и CH4.

Активным фронтом тактового сигнала при приеме данных в SDR и DDR режимах является возрастающий фронт. В режиме SDR отсчет сигнала принимается по возрастающему фронту, в режиме DDR отсчет начинается с возрастающего фронта.

Упаковка данных в параллельной шине АЦП в режиме DDR определяется следующими таблицами.

Таблица 21.161. Упаковка данных в DDR параллельной шине данных АЦП с мультиплексированием каналов

	DDR 16bit CH_mux MODE_OPT1=0	DDR 8bit CH_mux MODE_OPT1=1
CH1[15:0]	ADC_DATA [15:0] Rising_Edge	ADC_DATA [15:8] Rising_Edge выравнивание по старшим битам, младшие биты заполняются нулем
CH2[15:0]	ADC_DATA [15:0]	ADC_DATA [15:8] Falling_Edge

	Falling_Edge	выравнивание по старшим битам, младшие биты заполняются нулем
CH3[15:0]	0	ADC_DATA [7:0] Rising_Edge выравнивание по старшим битам, младшие биты заполняются нулем
CH4[15:0]	0	ADC_DATA [7:0] Falling_Edge выравнивание по старшим битам, младшие биты заполняются нулем

Таблица 21.162. Упаковка данных в DDR параллельную шину данных АЦП с мультиплексированием бит данных

	DDR 16bit bit_mux MODE_OPT1=0	DDR 16bit bit_mux MODE_OPT1=1
CH1[15:0]	Четные биты 14, 12, ….0: ADC_DATA [15:8] Rising_Edge Нечетные биты 15, 13, ….1: ADC_DATA [15:8] Falling_Edge	Нечетные биты 15, 13, ….1: ADC_DATA [15:8] Rising_Edge Четные биты 14, 12, ….0: ADC_DATA [15:8] Falling_Edge
CH2[15:0]	Четные биты 14, 12, ….0: DATA_OUT_R[7:0] Rising_Edge Нечетные биты 15, 13, ….1: ADC_DATA [7:0] Falling_Edge	Нечетные биты 15, 13, ….1: DATA_OUT_R[7:0] Rising_Edge Четные биты 14, 12, ….0: ADC_DATA [7:0] Falling_Edge
CH3[15:0]	0	0
CH4[15:0]	0	0

21.3.1.8.1.3 Формат принимаемых данных в CMOS шине АЦП в режиме JESD207 (AD9361 CMOS Full Duplex Dual Port Compatible)

Этот режим предназначен для ввода цифровых данных из RFFE типа AD9361 в режиме LVCMOS «Full Duplex Dual Port». При этом возможен как SDR, так и DDR режим. Количество каналов – один или два квадратурных. Максимальная разрядность данных – 15 бит. Младший нулевой бит шины данных задействован под строб RX_FRAME. При меньшей разрядности данных выходная шина АЦП должна подключаться к старшим разрядам шины, т.е. с выравниванием по MSB. При помощи поля OUT_DATA_BW задается актуальная ширина шины и количество младших разрядов, которые подвергаются обнулению.

Пример подключения 12 битного АЦП к интерфейсу представлен на Рисунок 21.20.

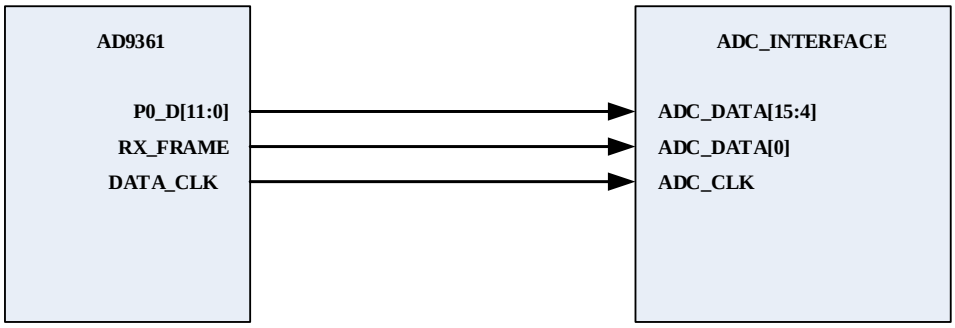


Рисунок 21.20. Пример подключения AD9361 в режиме CMOS JESD207 “Full Duplex Dual Port”

Интерфейс подразумевает передачу только квадратурных данных (I и Q компоненты). Количество каналов – 1 или 2. В интерфейсе принято следующее соответствие квадратурных каналов физическим

Канал интерфейса JESD207	Физический канал
CH1_I	CH1
CH1_Q	CH2
CH2_I	CH3
CH2_Q	CH4

21.3.1.8.1.4 Формат принимаемых данных в LVDS шине АЦП в режиме JESD207 (AD9364 LVDS Full Duplex Dual Port Compatible)

Этот режим предназначен для ввода цифровых данных из RFFE типа AD9361 в режиме LVDS «Full Duplex Dual Port». При этом возможен как SDR, так и DDR режим. Количество каналов – один или два квадратурных. Максимальная разрядность данных – 16 бит, разрядность шины данных 8 бит. Младший нулевой бит шины данных АЦП задействован под строб RX_FRAME. Выходная шина АЦП должна подключаться к старшим разрядам шины с выравниванием по MSB. При помощи поля OUT_DATA_BW осуществляется маскирование нулем незадействованных младших разрядов.

Схема подключения 12 битного АЦП к интерфейсу с шириной шины 6 бит представлена на Рисунок 21.21.

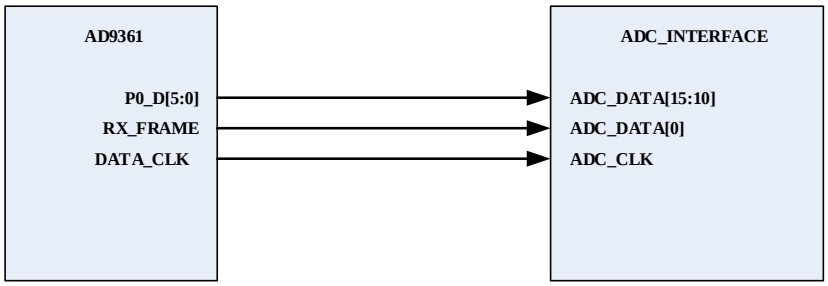


Рисунок 21.21. Пример подключения AD9361 в режиме LVDS “Full Duplex Dual Port”

Ниже на рисунках представлены временные диаграммы интерфейса для SDR и DDR передачи данных одного и двух каналов.

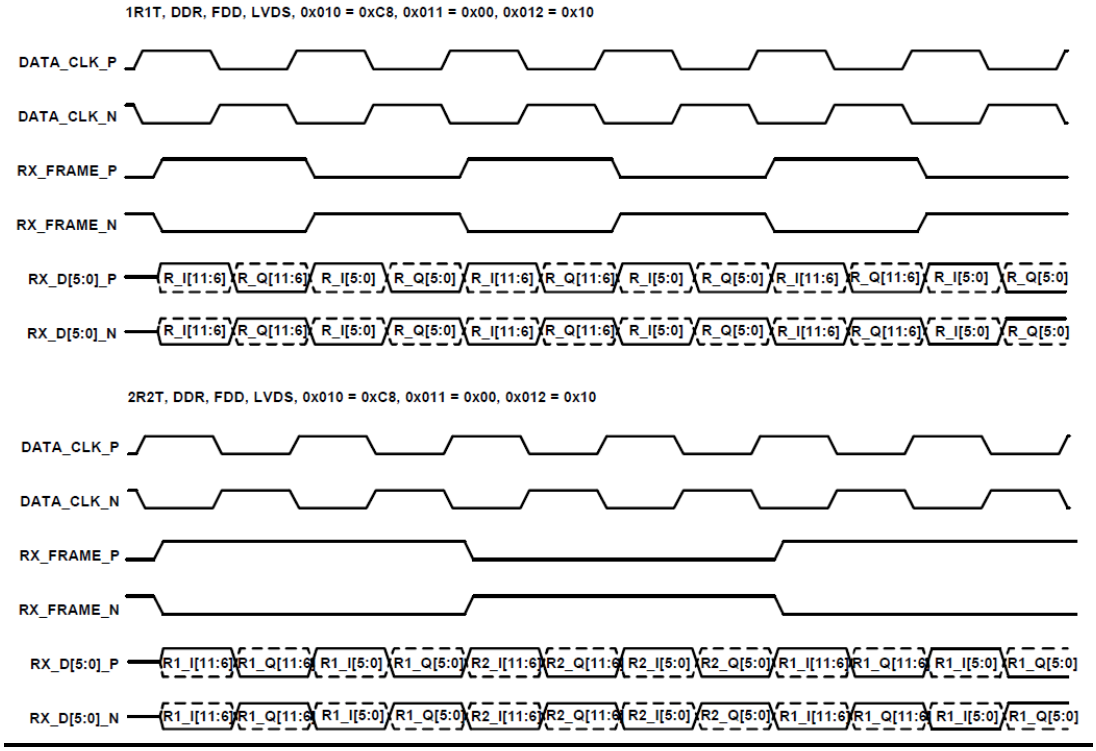


Рисунок 21.22. Временные диаграммы интерфейса AD9361 в режиме LVDS “Full Duplex Dual Port”

На диаграммах принято следующее соответствие каналов

На диаграмме	Физический канал
R_I, R1_I	CH1
R_Q, R1_Q	CH2

R2_I	CH3
R2_Q	CH4

Актуальная разрядность шины данных АЦП определяется полем MODE_OPT2. Поскольку в данном режиме каждый отсчет сигнала передается последовательно двумя половинками, разрядность сигнала на выходе интерфейса в два раза шире, чем количество задействованных битов шины. Последовательность передачи приведена на диаграмме выше.

21.3.1.8.1.5 Формат принимаемых данных в режиме JESD204

Формат данных соответствует стандарту. Каждый из четырех каналов JESD204 интерфейса соответствует одному из четырех выходных сигналов CH1…CH4 ADCINTERFACE.

21.3.1.8.1.6 Требование к тактовым частотам интерфейса.

Для корректной работы блока требуется, чтобы тактовая частота CLK была не ниже тактовой частоты ADC_CLK. В случае, когда тактовая частота CLK ниже, чем ADC_CLK происходит переполнение FIFO, расположенных на стыке частотных доменов, и вырабатывается сигнал ошибки CLK_ERR по событию переполнения FIFO интерфейса. При этом событие ошибки отражается формированием разового сигнала прерывания и защелкивается в поле CLK_ERR регистра ADC_CTRL. Сброс ошибки и повторная синхронизация осуществляются либо общим сбросом, либо полем регистра управления ADC_ENA.

21.3.1.8.2 Селектор источников входных сигналов - RX_IN_DTRT

Блок предназначен для коммутации потоков входных данных, получаемых от АЦП и формирования потоков данных для блоков обработки приемной части DFE. В блоке имеются следующие возможности обработки сигналов:

- формирование компонент аналитических сигналов из пар входных сигналов,
- формирование сигналов с расширенным динамическим диапазоном из пар входных сигналов от АЦП, преобразующих один и тот же аналоговый источник с разным коэффициентом усиления,
- подача тестовых последовательностей;

Блок имеет 4 входа разрядностью 16 бит, подключенных к выходам интерфейса АЦП. Для сигналов на каждом из 4 входов предусмотрена возможность оценки их параметров и коррекции смещения нуля.

Каждый из 4 выходов устройства разрядностью 2x26 бит подключен к своему приемному тракту RX_CHAN. Через каждый из выходов может выводиться сигнал от целого ряда источников. Схема коммутации выбора источника для одного из выходных каналов (первого) приведена на Рисунок 21.23.

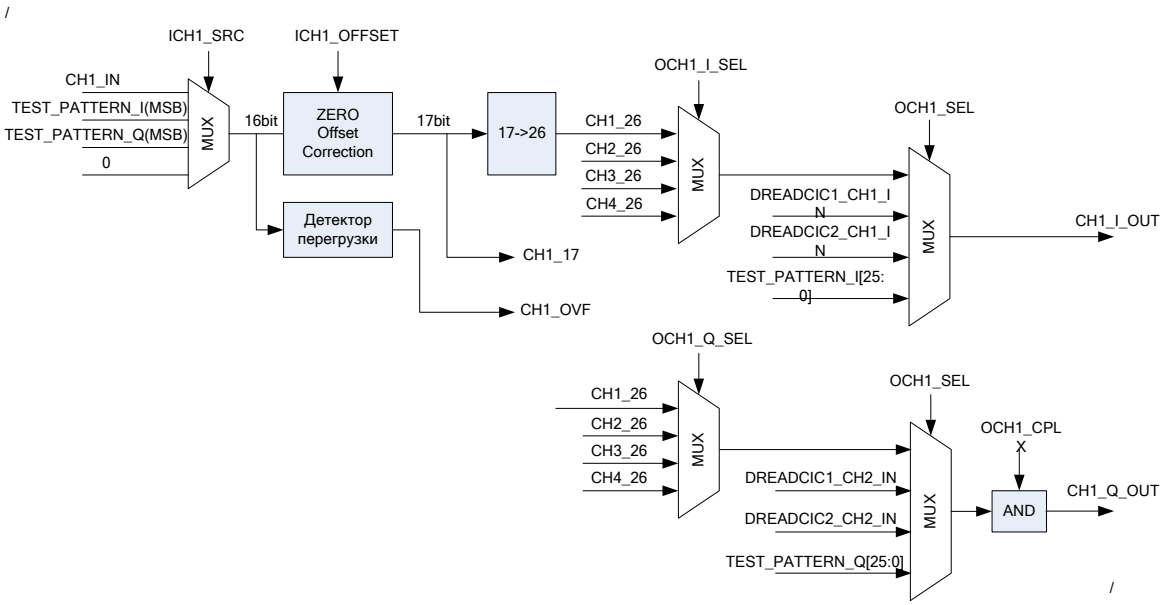


Рисунок 21.23. Блок-схема выбора источника для выходного канала 1

Символ «x» во всех наименованиях, перечисленных ниже, заменяется на номер выходного канала.

Сигналы выбора ICHx_SRC мультиплексоров на входе ICHx_SEL управляются одноименными полями регистра RX_IN_DTRT_ICH_CTRL.

Сигналы выбора OCHx_I_SEL, OCHx_Q_SEL и OCHx_SEL, а также сигнал включения комплексного режима OCHx_CPLX, управляются одноименными полями регистров RX_IN_DTRT_OCH12_CTRL для выходных каналов #1 и #2 и RX_IN_DTRT_OCH34_CTRL для выходных каналов #3 и #4.

При выборе режима действительного сигнала (поле OCHx_CPLX=0) Q-компонента выходного сигнала принудительно обнуляется.

21.3.1.8.2.1 Настройка коррекции смещения нуля

Коррекция смещения нуля в каналах осуществляется прибавлением константы к входным отсчетам. К принятому 16 разрядному отсчету прибавляется значение младших 16 разрядов регистра RX_IN_DTRT_INCH_OFFSETx. Разрядность сигнала на выходе корректора смещения нуля составляет 17 бит, что исключает переполнение в результате коррекции.

На входе корректора смещения нуля по каждому каналу осуществляется детектирование перегрузки в канале по условию равенства или превышения модулем сигнала значения, записанного в младшие 15 разрядов регистров RX_IN_DTRT_INCH_OVFTRSx. В случае установки нулевого значения регистра событие перегрузки не детектируется.

Для подбора оптимального значения параметра коррекции может быть использован блок оценки параметров сигнала.

21.3.1.8.2.2 Блок оценки параметров сигнала

Блок оценки параметров сигнала (Signal Estimator) встроен в селектор источников и имеет возможности для оценки следующих параметров сигнала:

- мощность;
- максимальное и минимальное значение;
- среднее значение сигнала.

Блок-схема Signal Estimator приведена на Рисунок 21.24.

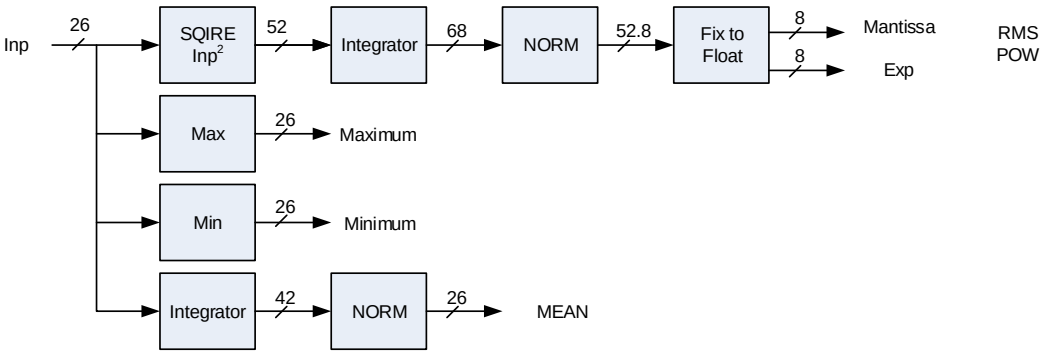


Рисунок 21.24. Блок-схема Signal Estimator

Оценка параметров сигнала производится на длительности выборки, задаваемой через поле SE_EST_DURATION регистра RX_IN_DTRT_TST_CTRL. Значение поля интерпретируется как степень числа 2, т.е. длительность выборки составляет $2^{SE\_EST\_DURATION}$ отсчетов.

Для оценки среднего значения (постоянной составляющей) и мощности сигнала производится деление его на кол-во накопленных отсчетов, реализованное как сдвиг вправо на величину SE_EST_DURATION.

Оценка мощности сигнала выводится в формате с плавающей точкой через поля MANT и EXP регистра RX_IN_DTRT_SE_POWRMS. Разрядности как мантиссы, так и экспоненты составляют 8 бит.

Минимум, максимум и оценка среднего значения могут быть прочитаны в младших 16 битах регистров, соответственно, RX_IN_DTRT_SE_MIN, RX_IN_DTRT_SE_MAX и RX_IN_DTRT_SE_MEAN.

Сигнал старта блока оценки сигнала формируется по факту записи единицы в поле SE_START_STATUS регистра RX_IN_DTRT_TST_CTRL. Если предыдущая процедура измерения не закончена, она прерывается.

По событию окончания измерений формируется сигнал прерывания INT_SE и устанавливается в «1» сигнал готовности, значение которого можно узнать, прочитав поле SE_START_STATUS регистра RX_IN_DTRT_TST_CTRL.

Блок оценки может быть подключен к различным сигналам внутри селектора. Выбор источника осуществляется через поле SE_SRC регистра RX_IN_DTRT_TST_CTRL и определен в таблице ниже.

Таблица 21.163. Коммутация потоков входных данных для блока Signal Estimator

SE_SRC	Анализируемый сигнал
0….3	CH1_I_OUT…. CH4_I_OUT
4….7	CH1_Q_OUT…. CH4_Q_OUT
8….11	Сигналы после компенсации постоянной составляющей: CH1_17 …. CH4_17
12…15	Зарезервировано

21.3.1.8.2.3 Генератор тестовых сигналов

В качестве тестового воздействия имеются 4 варианта: комплексная константа и три варианта действительного сигнала. Блок–схема коммутатора тестовых воздействий приведена на Рисунок 21.25.

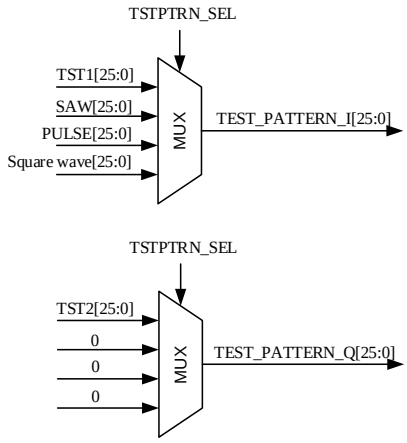


Рисунок 21.25. Селекция тестовых сигналов

Ниже в таблице приведены описания перечисленных тестовых воздействий

Таблица 21.164. Выбор типа тестового воздействия

TSTPTRN_SEL	Описание
0	Константы берутся из регистров управления TST1, TST2
1	«SAW» Пила. 26бит счетчик в двоично-дополнительном коде с инкрементом TST1. TST1 интерпретируется как знаковое. Начальное значение счетчика - ноль.
2	«Pulse». Импульс в один отсчет сигнала. По переходу SAW через ноль из отрицательного значения в положительное формируется положительный импульс амплитудой TST2. Начальное значение – ноль. По переходу SAW

TSTPTRN_SEL	Описание
	через ноль из положительного значения в отрицательное формируется отрицательный импульс амплитудой TST2. Начальное значение – ноль.
3	«Square». Меандр амплитудой TST2. По положительному импульсу устанавливается положительное значение, по отрицательному- отрицательное. Амплитуда формируется так же, как и для случая «Pulse»

Инкремент счетчика может быть положительным и отрицательным. Тем самым изменяется как направление зубцов пилообразного сигнала, так и полярность первого импульса и меандра. Генератор тестовых воздействий приводится в исходное состояние входным сигналом SYNC_START=0 или стробом SYNC_CFG.

Ниже на Рисунок 21.26 приведены временные диаграммы тестовых сигналов

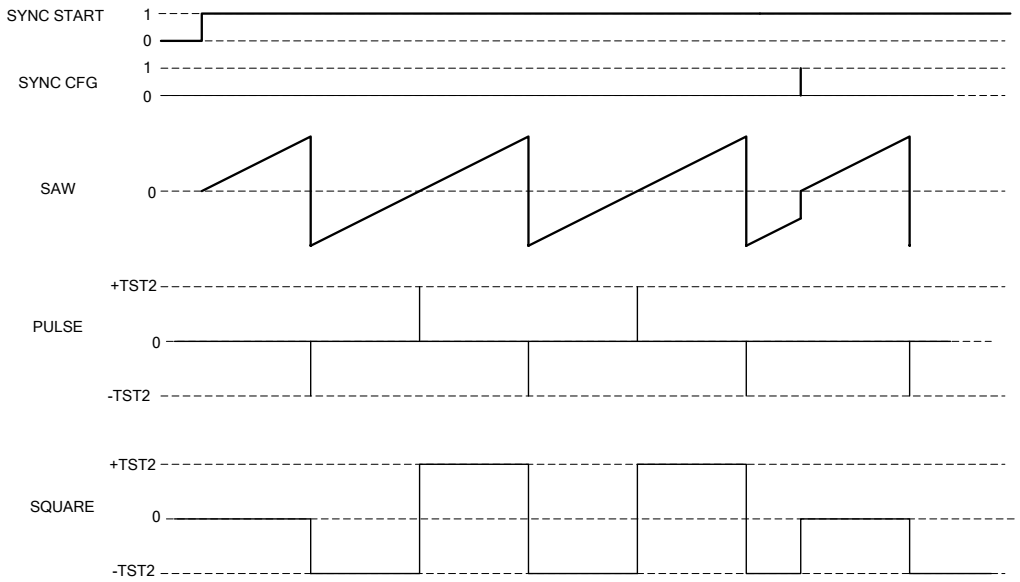


Рисунок 21.26. Временные диаграммы тестовых сигналов

Сформированные тестовые сигналы TEST_PATTERN_I и TEST_PATTERN_Q могут быть поданы в качестве воздействий в разные точки обработки сигнала внутри коммутатора, как описано на Рисунок 21.23.

21.3.1.8.3 DREADCIC - блок расширения динамического диапазона АЦП и компенсации небаланса каналов I/Q

У данного блока есть два назначения, преследующих разные цели, но требующих схожих операций над отсчетами сигналов. Они реализованы как два режима работы:

- Режим расширения динамического диапазона АЦП и предотвращения его перегрузки для действительного выходного сигнала;
- Компенсация небаланса квадратур и фаз для комплексного выходного сигнала.

Поскольку в модуле используются операции умножения, реализованные в виде конвейера, блок вносит задержку в несколько тактов в процессе обработки сигнала. Выходные сигналы синхронизации формируются с привязкой к отсчету и задерживаются строго на то же время, что и сигнал.

Сигнал синхронизации старта используется блоком как сигнал разрешения работы. При SYNC_START=0 тракт обработки сигнала приводится в нулевое состояние, работа блока запрещена.

Блок управляется следующими регистрами (символ x в наименовании заменяется на номер устройства):

Таблица 21.165. Перечень регистров DREADCIC

Название	Тип	Бит	Описание
RX_IN_DTRT_DREADCIC_CTRL	R/W	16	Регистр коммутации входов и включения
DRE_ADCICx_CONTROL_STATUS	R/W	16	Регистр управления и статуса DRE_ADCIC.
DRE_ADCICx_GAIN_ERR	R/W	16	Выходной сигнал ошибки оценки усиления
DRE_ADCICx_OFFSET_ERR	R/W	16	Выходной сигнал ошибки смещения нуля.
DRE_ADCICx_K1	R/W	16[14]	Двоичное положительное число 14 бит В режиме DRE: дробная часть коэффициента усиления для канала DATA_IN_CH1 В режиме ADCIC: усиление в канале A для компенсации амплитудного небаланса.
DRE_ADCICx_K2	R/W	16[13]	Двоичное знаковое число 13 бит В режиме DRE: целая часть коэффициента усиления для канала DATA_IN_CH1, младшие 8 бит В режиме ADCIC – усиление в канале A для компенсации фазового небаланса, младшие 13 бит
DRE_ADCICx_K3	R/W	16[14]	Двоичное положительное число В режиме DRE не используется. В режиме ADCIC – усиление в канале B для компенсации амплитудного небаланса, младшие 14 бит
DRE_ADCICx_GAIN_INT	R	16[8]	Двоичное положительное число Возвращает актуальное значение целой части коэффициента усиления.
DRE_ADCICx_GAIN_FRC	R	16[14]	Двоичное положительное число Возвращает актуальное значение дробной части коэффициента усиления.

Обобщенная блок-схема DRE_ADCIC приведена на Рисунок 21.27

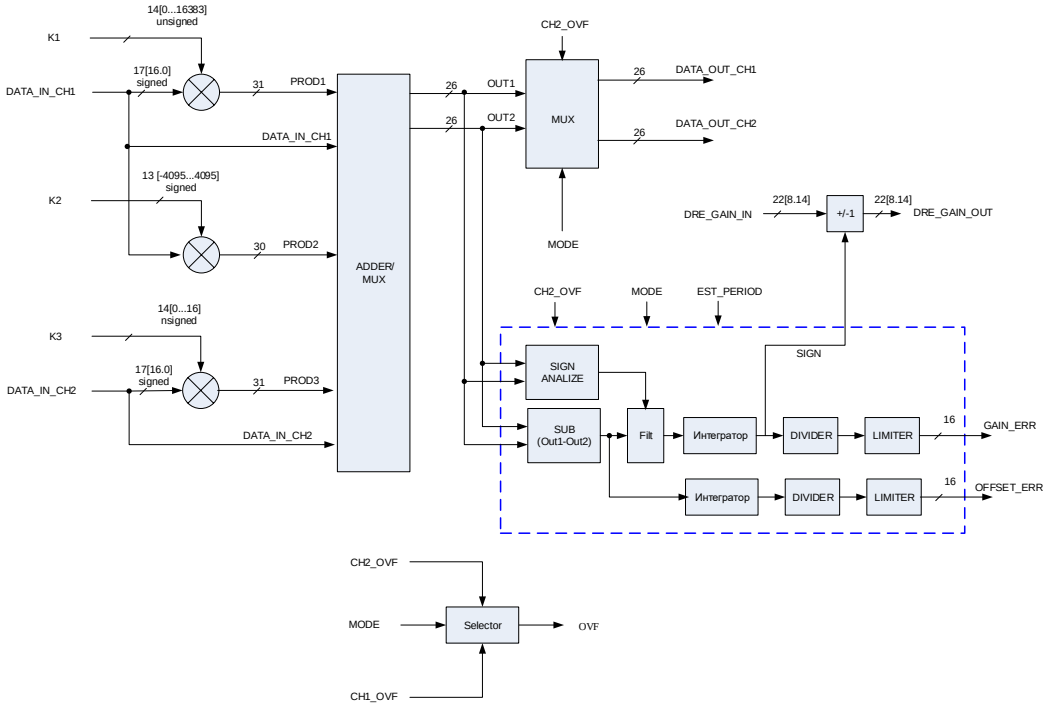


Рисунок 21.27. Блок-схема DRE_ADCIC

Поле DRE_ADCIC_ENA регистра DRE_ADCICx_CONTROL_STATUS разрешает работу блока. При DRE_ADCIC_ENA=0 работа блока запрещена. Управляющие регистры K1, K2, K3 прозрачны – не требуется подавать команду на обновление через запись в поле KOFF_UPDATE. Тракт обработки сигнала сброшен.

Включение подачи сигналов и их коммутация осуществляются через поля регистра RX_IN_DTRT_DREADCIC_CTRL. Поля DREADCIC1_ENA и DREADCIC2_ENA включают соответствующий блок при записи «1». Четыре поля DREADCICx_CHy_SEL разрядностью по 2 бита управляют источником сигнала входа y устройства x. Схема коммутации показана на примере устройства DREADCIC1 на Рисунок 21.28.

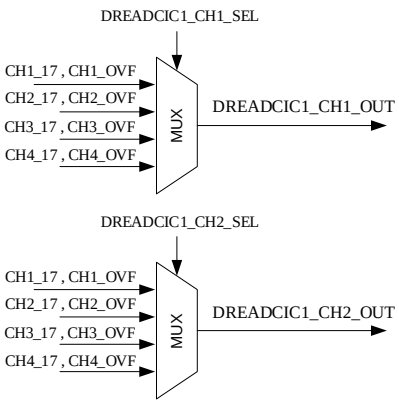


Рисунок 21.28. Селекция входных сигналов для блока DREADCIC1

В качестве входных сигналов для блоков DRE_ADCIC используются 17 разрядные сигналы данных с выхода блока коррекции нуля CH_17 и сопровождающие их сигналы признака перегрузки CHx_OVF.

21.3.1.8.3.1 Режим расширения динамического диапазона DRE

Входной сигнал блока – сигналы двух АЦП после коррекции смещения нуля, преобразующие один и тот же аналоговый действительный сигнал с разными коэффициентами передачи. Сигнал с большим коэффициентом передачи следует подавать на вход DATA_IN_CH2 – основной рабочий канал в режиме малого сигнала. При отсутствии перегрузки (CH2_OVF=0) этот сигнал расширяется знаком до 26 бит и передается на выход блока. В случае детектирования перегрузки в основном канале DATA_IN_CH2 (CH2_OVF=1), выходной сигнал формируется масштабированием DATA_IN_CH1.

Масштабирование 17-битного сигнала DATA_IN_CH1 выполняется его умножением на 20 разрядный положительный коэффициент в представлении 8.14 (8 бит целая часть, 14 бит-дробная). Тем самым достигается сшивка передаточных характеристик двух каналов на границе динамического диапазона канала CH2. В целях экономии разрядности умножителей, умножение реализовано как сумма произведений отсчета сигнала на дробную и целую части, что отражено на схеме. Дробная часть коэффициента задается в младших 14 битах регистра DRE_ADCICx_K1, целая часть – в младших 8 битах регистра DRE_ADCICx_K2. Результат умножения представляет собой знаковое число в формате 26.14 (26 бит знаковая целая часть, 14 бит – дробная часть), от которого берется только целая часть и подается на выход OUT1 блока ADDER/MUX. Выходной сигнал OUT2 того же блока – знакорасширенный до 26 бит сигнал DATA_IN_CH2.

Выходной сигнал DATA_OUT_CH1 равен OUT2 в отсутствии признака перегрузки в канале CH2. В противном случае равен OUT1.

Выходной сигнал DATA_OUT_CH2 всегда равен нулю.

Режим автокоррекции.

В режиме DRE может быть задействован имеющийся в блоке автомат коррекции коэффициента усиления, который включается установкой в «1» поля DRE_AUTO_GAIN_ADJ регистра DRE_ADCICx_CONTROL_STATUS. В режимах, отличных от DRE, автомат выключен и не генерирует никаких сигналов.

При работе автомата коэффициент усиления корректируется по результатам измерения ошибки баланса передаточных характеристик двух каналов. Стартовое значение коэффициента масштабирования, так же, как и в ручном режиме, записывается в регистры DRE_ADCICx_K1 и DRE_ADCICx_K2. По команде старта коррекции, передаваемой как запись «1» в поле KOFF_UPDATE регистра DRE_ADCICx_CONTROL_STATUS, коэффициент актуализируется. Значения целой и дробной части актуального коэффициента

передачи отображаются в регистрах DRE_ADCICx_GAIN_INT и DRE_ADCICx_GAIN_FRC.

Ошибка баланса усиления каналов оценивается путем усреднения по времени результата вычисления выражения sign(OUT2)*(OUT1–OUT2). Усреднение производится на количестве отсчетов, задаваемом в поле EST_PERIOD регистра DRE_ADCICx_CONTROL_STATUS. Длительность интервала составляет $2^{(EST\_PERIOD+4)}$ отсчетов сигнала. При усреднении учитываются только отсчеты, не сопровождаемые сигналом перегрузки и имеющие одинаковый знак в обоих каналах. Остальные отсчеты из анализируемой выборки исключаются и счетчик отсчетов не инкрементируется, т.е. реальное время оценки может оказаться больше, чем $2^{(EST\_PERIOD+4)}$ отсчетов.

После накопления результат сдвигается вправо на количество разрядов EST_PERIOD, что эквивалентно делению на количество накопленных отсчетов $2^{(EST\_PERIOD+4)}$. Если после сдвига накопленное значение не равно нулю, производится коррекция актуального коэффициента усиления в нужную сторону путем добавления или вычитания единицы из дробной части коэффициента. Операция производится с переносом в целую часть и контролем переполнения. Таким образом, после нескольких процедур автоматической коррекции баланс в каналах восстанавливается.

Последнее значение оценки ошибки баланса усиления может быть прочитано через регистр DRE_ADCICx_GAIN_ERR.

Процедура коррекции может быть запущена как однократно, по прямой команде, так и автоматически по окончании предыдущей процедуры.

Установка бита EST_AUTOREPEAT регистра DRE_ADCICx_CONTROL_STATUS включает режим бесконечного автоматического повторения процедуры коррекции и запускает первую итерацию, если устройство свободно. Изменение состояния данного бита не влияет на текущую процедуру, если она осуществляется в момент изменения.

При EST_AUTOREPEAT=0 может быть произведен однократный запуск процедуры коррекции, инициируемый записью «1» в поле EST_STATUS. Процедура стартует, если устройство свободно. В противно случае, если процедура уже осуществляется, запись единицы в поле EST_STATUS игнорируется, а запись нуля прерывает процедуру. При EST_AUTOREPEAT=1 любые записи в поле EST_STATUS игнорируются.

Каждый раз при окончании процедуры значение оценок записывается в регистры DRE_ADCICx_GAIN_ERR и DRE_ADCICx_OFFSET_ERR, применяется содержимое поля EST_PERIOD, если оно было изменено во время процедуры и вырабатывается сигнал прерывания EST_RDY.

Состояние автомата коррекции отображается в поле EST_STATUS. При старте поле сбрасывается, по окончании процедуры поле устанавливается.

Сигнал перегрузки OVF в режиме DRE вырабатывается по сигналу перегрузки в канале с меньшим коэффициентом усиления (вход DATA_IN_CH1).

Оценка смещения между каналами.

При активности процедуры коррекции, параллельно с ней производится оценка взаимного смещения двух каналов путем накопления разности OUT1-OUT2 в отсутствии перегрузки в канале CH2. Результат накопления нормируется сдвигом вправо на количество разрядов EST_PERIOD. Последнее значение оценки сдвига между каналами может быть прочитано через регистр DRE_ADCICx_OFFSET_ERR.

Данная оценка необходима, т.к. из-за масштабирования путем умножения на коэффициент смещение нуля в канале OUT1 также возрастает пропорционально коэффициенту. Точности коррекции нуля в канале АЦП с меньшим усилением не хватает для удержания смещения в канале OUT1 в пределах младших одного-двух разрядов. Однако баланс смещения между каналами можно восстановить, скорректировав смещение в канале OUT2, который не подвергается масштабированию. В этом случае повышается качество сшивки двух каналов, уменьшаются скачки уровня при переключении между ними, но достигается это ухудшением точности коррекции смещения нуля на выходе устройства.

21.3.1.8.3.2 Режим компенсации небаланса АЦП (ADCIC)

Данный режим применяется при работе с парами входных сигналов, полученных от АЦП, преобразующих сигналы пары компонент комплексного аналогового сигнала. Устройство позволяет компенсировать ошибки коэффициента усиления компонент сигнала и сдвига фазы между ними, возникающие в аналоговом тракте. Ошибка усиления компенсируется прибавлением к компоненте ее копии, умноженной на коэффициент. Ошибка сдвига фазы между компонентами, который в норме должен составлять 90 градусов, компенсируется добавлением части компоненты, принимаемой каналом DATA_IN_CH1 к каналу DATA_IN_CH2. Вычисление корректирующих коэффициентов осуществляется на основе априорных данных о трактах обработки и анализе сигналов, принимаемых на входах интерфейса АЦП. Средства автоматической коррекции для данного режима не предусмотрены.

Корректирующие коэффициенты являются целыми числами со знаком и задаются младшими битами регистров DRE_ADCICx_K1[13:0], DRE_ADCICx_K2[12:0] и DRE_ADCICx_K3[13:0]. Данные, записанные в регистры, применяются не сразу после записи, а по команде на обновление подаваемой записью «1» в поле KOFF_UPDATE регистра DRE_ADCICx_CONTROL_STATUS. Этим достигается синхронность их обновления.

Отсчеты сигналов OUT1 и OUT2 вычисляются по формулам:

OUT1 = (DATA_IN_CH1 << 6) + (K1*DATA_IN_CH1 >> 8);

OUT2 = (DATA_IN_CH2 << 6) + ((K3*DATA_IN_CH2 + K2*DATA_IN_CH1) >> 8).

Перед операцией сдвига влево разрядность отсчета сигнала расширяется до 26 бит добавлением старших битов с расширением знака. При сдвиге влево младшие разряды заполняются нулями.

Результаты вычисления произведений имеют разрядность 31 бит. Перед сдвигом их разрядность расширяется добавлением старших битов с расширением знака до 34 бит. При Сдвиг вправо реализован отсечением младших битов с последующим округлением методом прибавления самого старшего из отброшенных разрядов. Результат имеет разрядность 26 бит.

Сложение результатов сдвига производится при разрядности 26 бит. Переполнение при всех описанных операциях исключено за счет избыточности разрядности.

Сигнал перегрузки OVF в режиме ADCIC вырабатывается по логическому «или» от двух сигналов перегрузки входных сигналов.

21.3.1.8.4 Приемный канал цифровой обработки сигнала - RX_CHNL

Приемный тракт DFE_ALT включает в себя 4 канала обработки сигнала. Каждый канал RX_CHNL осуществляет перенос частоты, децимацию и канальную фильтрацию одного цифрового квадратурного сигнала.

Каждый канал управляется регистрами, доступными при включении нужной страницы в диапазоне адресов 0xA000-0xAFFF. Выбор режима адресации и страницы осуществляется через регистр RX_CTRL. Доступен групповой режим адресации, при котором управляющие данные записываются одновременно в несколько каналов, но чтение данных невозможно.

Общее управление каналом доступно через регистр RX_CHNL_CTRL.

Битом ENABLE можно включить (1) и выключить (0) канал.

Бит STATE отражает состояние синхронизации: «0» – ожидание SYNC_START, «1» – прием данных и обработка.

Бит SYNC_EVENT устанавливается при достижении сигналом SYNC_CFG выхода канала.

Бит PPS_EVENT устанавливается при возникновении события PPS.

Поле OVF_STATUS является 8-битным составным индикатором перегрузок. Каждый его бит отражает статус перегрузки определенного устройства (см. описание регистра).

Биты SYNC_EVENT, PPS_EVENT и биты поля OVF_STATUS сбрасываются записью в них значения «1».

Обобщенная блок-схема RX_CHNL приведена на Рисунок 21.29.

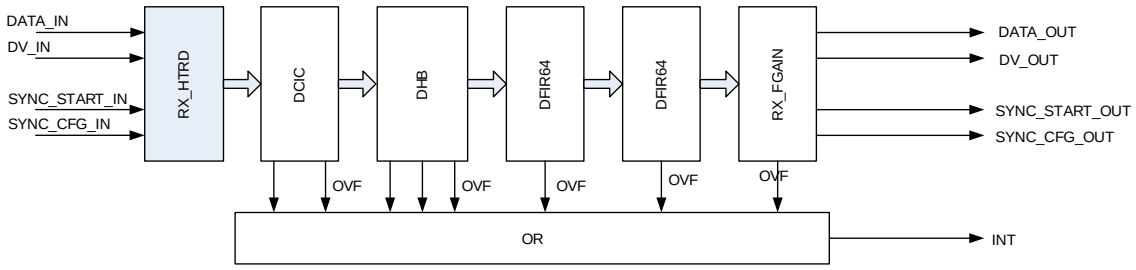


Рисунок 21.29. Блок-схема канала приема RX_CHNL

В канал последовательно включены следующие блоки обработки:

- Блок переноса частоты с функцией свипа;
- Двухкаскадный CIC фильтр-дециматор;
- Трехкаскадный полуполосный фильтр-дециматор;
- Два КИХ фильтра с произвольными коэффициентами;
- Устройство масштабирования и коррекции фазы.

Разрядность данных межблочных соединений составляет 2x26 бит. Выходы сигналов прерываний по перегрузкам блоков объединяются операцией «или» в единый выходной сигнал прерывания от канала.

При непрерывной работе приемника в каждом приемном канале производится подсчет принятых отсчетов на 32-разрядном счетчике, работающем с переполнением. В момент возникновения события PPS значение счетчика копируется в регистр RX_CHNL_PPS_POS. При выключении приемника по обнулению SYNC_START счетчик сбрасывается. Таким образом, анализируя данный регистр, можно узнать точное положение последнего события PPS относительно начала работы блока по сигналу SYNC_START=1. С момента старта канала и до первого события PPS данных о времени прихода PPS еще нет, и из регистра читается сигнатура 0xFFFFFBAD. Поскольку данное значение может быть достигнуто только через несколько секунд работы канала на самой высокой частоте дискретизации, его детектирование в течение первых двух секунд свидетельствует о сбое генератора PPS. Более надежным способом подтверждения валидности счетчика является проверка битового поля PPS_EVENT регистра RX_CHAN_CTRL, которое устанавливается в «1» по возникновению события PPS.

21.3.1.8.4.1 Гетеродин RX_HTRD

Блок гетеродина представляет собой синтезатор частоты, работающий по принципу бесконечного накопления фазы в интеграторе, работающем без контроля переполнения (overlap), дополненным алгоритмом поворота фазы сигнала CORDIC. Также, в гетеродине реализована функция изменения частоты по кусочно-линейному закону – режим свип.

Блок управляется следующими регистрами:

Таблица 21.166. Перечень регистров RX_HTRD

Название поля [значение по сбросу]	Тип	Бит	Описание
RX_NCO_FRQ [0x0]	Read/Write	32	Частота гетеродина.
RX_NCO_SWRATE [0x0]	Read/Write	32	Приращение частоты гетеродина
RX_NCO_SWTIME [0x7FFF FFFF]	Read/Write	32	Период свипа.
RX_NCO_PHASE [0x0]	Read/Write	16	Смещение фазы гетеродина.
RX_NCO_MODE	Read/Write	16	Режим работы гетеродина

Обобщенная блок-схема квадратурного гетеродина приведена на Рисунок 21.30.

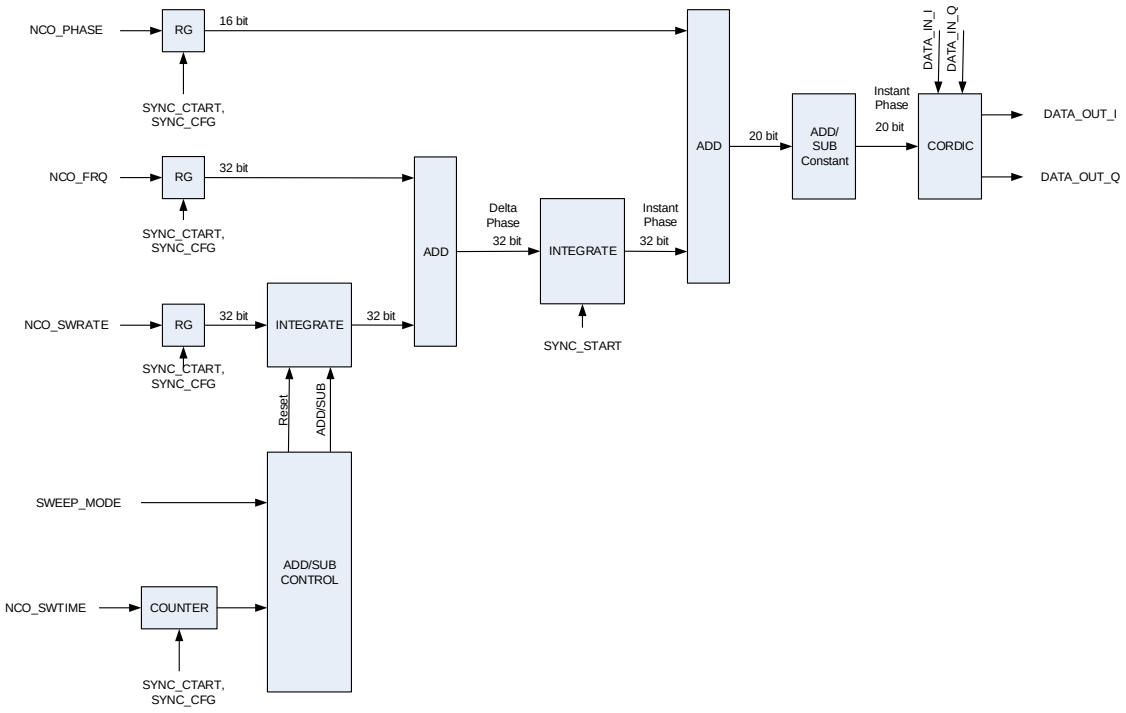


Рисунок 21.30. Блок-схема квадратурного гетеродина

Конфигурирование блока может осуществляться в двух режимах – прозрачном, когда новые значения актуализируются немедленно после записи, и в синхронном, когда актуализация происходит по сигналу синхронизации SYNC_CFG. Режим выбирается полем

REG_TRANSPARENT регистра RX_NCO_MODE. При REG_TRANSPARENT=1, а также при SYNC_START=0 включен прозрачный режим.

Режим постоянной частоты.

При выключенном режиме свипа (в регистре RX_NCO_MODE поле SWEEP_MODE=0) частота гетеродина определяется только значением, заданном в регистре RX_NCO_FRQ. Значение поля для задания нужной частоты можно вычислить по формуле

$$RX\_NCO\_FRQ = round(2^{32}*(Fh/Fs)),$$

где Fs — частота дискретизации сигнала на входе гетеродина,

Fh — требуемая частота гетеродина.

Основной интегратор фазы гетеродина на каждом отсчете сигнала прибавляет к своему аккумулятору значение, заданное в регистре RX_NCO_FRQ.

Сброс интегратора осуществляется при SYNC_START=0. В этом случае, на выходе генератора гетеродина устанавливается начальное значение фазы, задаваемое через регистр RX_NCO_PHASE. Значение данного регистра интерпретируется как дробное число в диапазоне $0..1-2^{-32}$, при умножении которого на 2π можно получить фазу, выраженную в радианах.

Режим свипа.

В данном режиме частота гетеродина, заданная в регистре RX_NCO_FRQ, получает смещение, которое изменяется по кусочно-линейному закону. График изменения частоты представляет собой «пилу». Смещение частоты вычисляется интегратором блока управления свипом.

Значение RX_NCO_SWRATE является значением приращения частоты за один отсчет сигнала и является знаковым двоично-дополнительным числом. Оно задает знак и скорость изменения частоты и имеет ту же размерность, что и RX_NCO_FRQ. Т.е. RX_NCO_SWRATE можно вычислить по похожей формуле:

$$RX\_NCO\_ SWRATE = round(2^{32}*(Fsw/Fs)),$$

где Fs — частота дискретизации сигнала на входе гетеродина,

Fsw — требуемое изменение частоты за 1 отсчет сигнала.

Значение RX_NCO_SWTIME является периодом между сменами режима приращения частоты, выраженном в отсчетах сигнала и является беззнаковым числом. Оно используется как предельное значение счетчика отсчетов, по достижении которого последний обнуляется и выдает синхронизирующий импульс. Синхронизация начала периода свипа достигается обнулением счетчика при SYNC_START=0 или по сигналу SYNC_CFG при установленном в «1» бите SWEEP_SYNC регистра RX_NCO_MODE.

Алгоритмы переключения режима изменения частоты в зависимости от режима свипа приведены в таблице.

Таблица 21.167. Режимы свипа частоты RX_HTRD

SWEEP_MODE	Тип	Реакция на импульс окончания периода свипа
00	выключено	-
01	SAW 1	Изменение направления приращения частоты на каждом периоде. Порядок работы по последовательным периодам: Старт – инкремент – декремент – инкремент – декремент…
01	SAW 2	Изменение направления приращения частоты на каждом втором периоде, начиная со второго после старта. Порядок работы по последовательным периодам: Старт – инкремент – декремент – декремент – инкремент…
11	SAW 3	Сброс интегратора по импульсу от счетчика.

Рисунок 21.31 представлены временные диаграммы изменения частоты в различных режимах свипа.

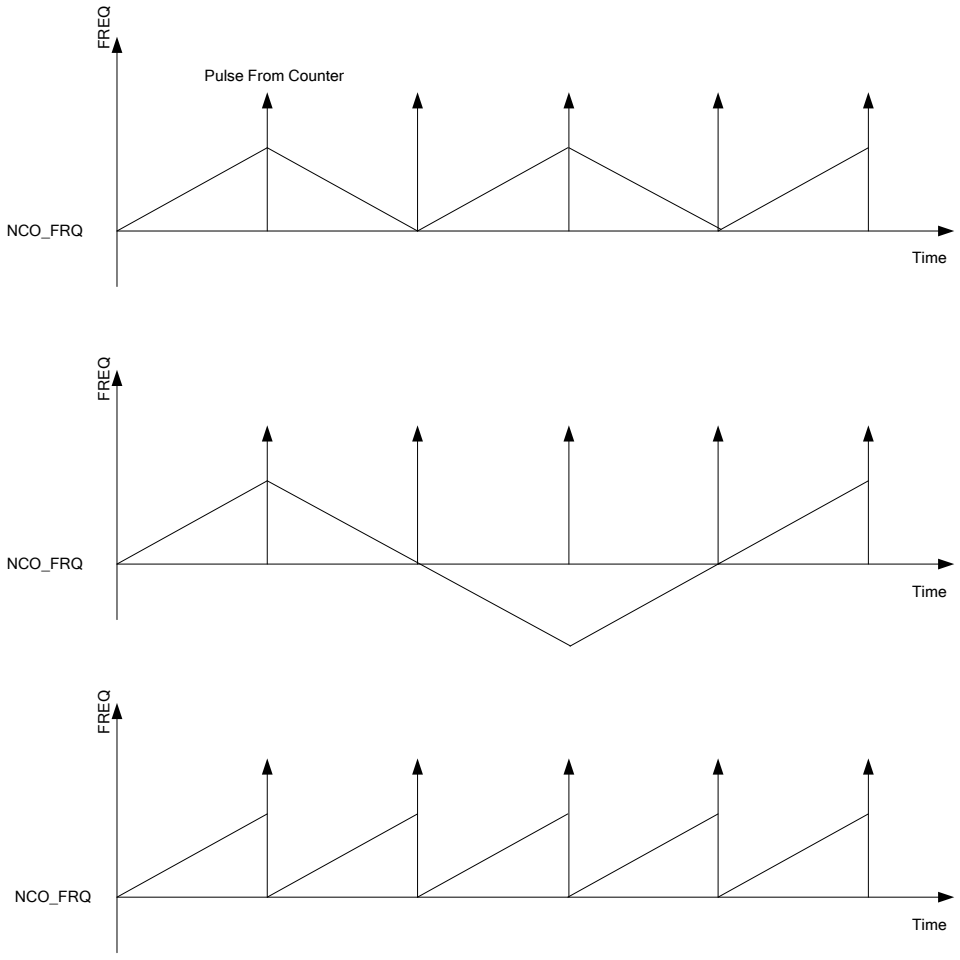


Рисунок 21.31. Временные диаграммы режимов свипа частоты RX_HTRD

Интегратор частоты осуществляет прибавление или вычитание (зависит от режима) к собственному аккумулятору значения регистра RX_NCO_SWRATE на каждом отсчете сигнала. Переполнение аккумулятора интегратора не контролируется (overlap).

Сброс интегратора происходит по тем же условиям, что и сброс счетчика периода свипа, т.е. при SYNC_START=0 и по приходу импульса SYNC_CFG при установленном в «1» бите SWEEP_SYNC регистра RX_NCO_MODE. Это означает, что при старте и смене режимов свипа изменение частоты всегда начинается со значения, установленного в регистре RX_NCO_FRQ.

Преобразование входного сигнала.

Фаза, полученная после сложения значения на выходе основного интегратора со значением смещения, заданном в регистре RX_NCO_PHASE, ограничивается по разрядности до 20 бит с округлением. Полученная мгновенная фаза поступает синхронно с отсчетами сигнала на блок координатного вращения CORDIC, который осуществляет поворот фазы сигнала на заданный угол. Поскольку данный блок реализован как конвейер, он вносит задержку в обрабатываемый сигнал. Сигнал SYNC_CFG_OUT задерживается синхронно с обрабатываемыми данными.

21.3.1.8.4.2 DCIC дециматоры

Блок осуществляет первую стадию децимации сигнала работая с сигналом самой высокой частоты, что обусловило применение в нем наиболее просто реализуемых фильтров – CIC, Cascaded Integrator Comb, не требующих применения устройств умножения.

Обобщенная блок-схема DCIC дециматора приведена на Рисунок 21.32.

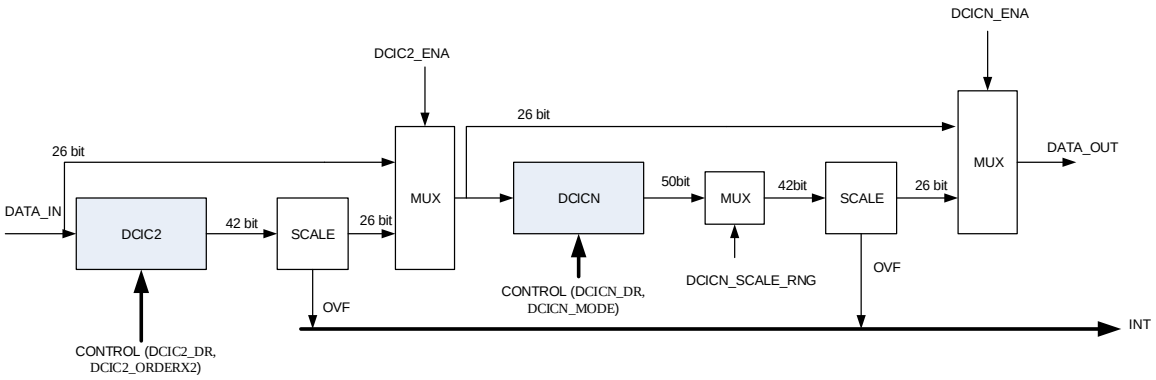


Рисунок 21.32. Блок-схема DCIC дециматора

Блок включает в себя два фильтра-дециматора – DCIC2 и DCICN. После каждого из них включено устройство масштабирования, приводящее разрядность сигнала к общему ограничению тракта обработки в 26 бит.

Первый фильтр DCIC2 является двухкаскадным CIC дециматором с максимальным коэффициентом децимации 128. Он управляется совместно с первым устройством масштабирования через поля регистра DCIC2_CFG.

Коэффициент децимации выбирается полем DCIC2_DR, в которое записывается уменьшенное на 1 его значение. При записи «0» фильтр выключается и переходит в режим bypass вместе со своим устройством масштабирования.

С помощью поля DCIC2_ORDERX2 порядок фильтра может быть выбран равным коэффициенту децимации (DCIC2_ORDERX2=0) или его удвоенному значению (DCIC2_ORDERX2=1).

Коэффициент масштабирования выбирается полем DCIC2_SCALE. Поле определяет, на сколько бит будет сдвинут влево 42 разрядный сигнал, взятый с выхода DCIC2, прежде чем от него будут взяты старшие 26 разрядов. В случае переполнения при сдвиге устанавливается бит DCIC2 в поле OVF_STATUS регистра RX_CHNL_CTRL.

Второй каскад DCICN реализован как CIC дециматор с переключаемым количеством каскадов от 3 до 7. Максимальный коэффициент децимации в каскаде CICN зависит от конфигурации и ограничен величинами:

- для CIC4 в 128 раз
- для CIC4 в 64 раза,
- для CIC5 в 27 раз,
- для CIC6 в 16 раз.

Фильтр-дециматор DCICN управляется совместно со вторым устройством масштабирования через поля регистра DCICN_CFG.

Коэффициент децимации выбирается полем DCICN_DR, в которое записывается уменьшенное на 1 его значение. При записи «0» фильтр выключается и переходит в режим bypass вместе со своим устройством масштабирования.

Поле DCICN_MODE определяет количество задействованных каскадов, которое устанавливается равным DCICN_MODE+3.

Масштабирование управляется полями DCICN_SCALE_RNG и DCICN_SCALE. Первое из них управляет выбором 42-битного диапазона из 50-битного сигнала с выхода DCICN. При установленном в «1» поле выбираются младшие 42 бита, в противном случае – старшие. Второе управляет сдвигом так же, как и в случае DCIC2.

При записи единицы в поле CLR регистра DCIC2_CFG или DCICN_CFG происходит сброс счетчика децимации и очистка данных в тракте вычислений связанного с ним устройства.

К выходу каждого мультиплексора подключен блок оценки уровня сигнала BWGAUGE. BWGAUGE анализирует заполненность разрядной сетки – количество задействованых бит на интервале анализа. Полученные оценки уровня сигнала отображаются в регистре DCIC_BWGAUGE. Операция записи в регистр DCIC_BWGAUGE приводит к сбросу

оценок и началу нового цикла оценки. Оценка – положительное целое число, указывающее на первый значащий бит.

21.3.1.8.4.3 Полуполосные фильтры-дециматоры DHB

Блок полуполосных фильтров состоит из трех включенных последовательно КИХ фильтров-дециматоров с постоянными коэффициентами DHB1, DHB2 и DHB3, каждый из которых может осуществлять децимацию в 2 раза либо работать в прозрачном режиме (bypass). На выходе каждого фильтра установлено устройство масштабирования для приведения разрядности к стандартной величине 26 бит. Все перечисленные устройства управляются через регистр DHB_MODE. При записи единицы в поле CLR регистра управления происходит сброс счетчика децимации и очистка данных в тракте вычислений.

Поле DHB_MODE включает и выключает каскады. Фильтры включаются последовательно, начиная с последнего. Т.е. нельзя включить DHB2, не включив DHB3, и нельзя включить DHB1, не включив DHB2 и DHB3.

Поля DHB3_SCALE, DHB2_SCALE и DHB1_SCALE управляют устройствами масштабирования. Все они функционируют одинаково, задавая величину сдвига влево в диапазоне 0..7 перед взятием старших 26 битов из 42 бит выхода фильтра.

К выходу каждого устройства масштабирования подключен блок оценки уровня сигнала BWGAUGE. Полученные оценки уровня сигнала отображаются в регистре DHB_BWGAUGE. Операция записи в регистр DHB_BWGAUGE приводит к сбросу оценок и блоков BWGAUGE и началу нового цикла оценки.

Коэффициенты фильтров приведены в таблице.

Таблица 21.168. Коэффициенты DHB фильтров

DHB1	DHB2	DHB3
		7
		0
		-67
		0
		348
		0
		-1281
		0
	61	3736
	0	0
	-564	-9252
	0	0
403	2681	20757
0	0	0
-3256	-9364	-46970
0	0	0
19237	39954	163794
32768	65536	262144
19237	39954	163794
0	0	0
-3256	-9364	-46970

DHB1	DHB2	DHB3
0	0	0
403	2681	20757
	0	0
	-564	-9252
	0	0
	61	3736
		0
		-1281
		0
		348
		0
		-67
		0
		7

АЧХ фильтров представлены на рисунках ниже.

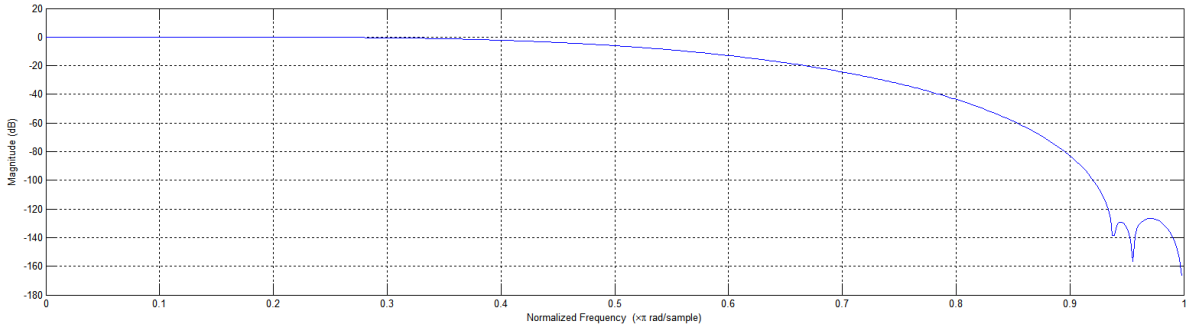


Рисунок 21.33. АЧХ DHB1 фильтра

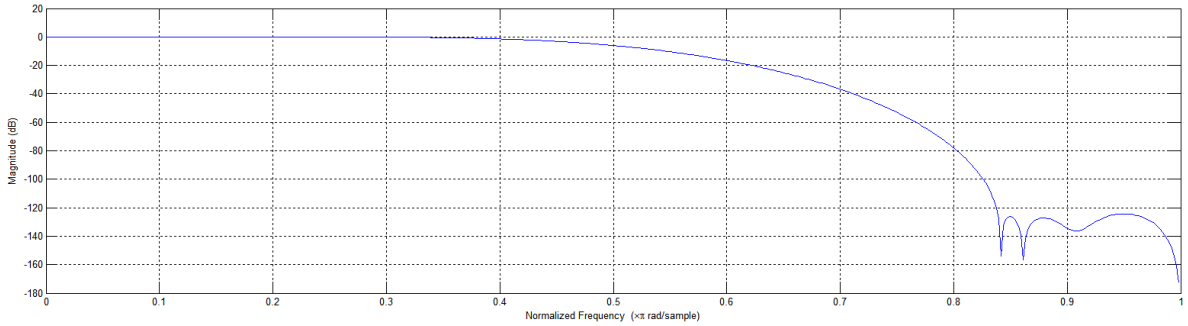


Рисунок 21.34. АЧХ DHB2 фильтра

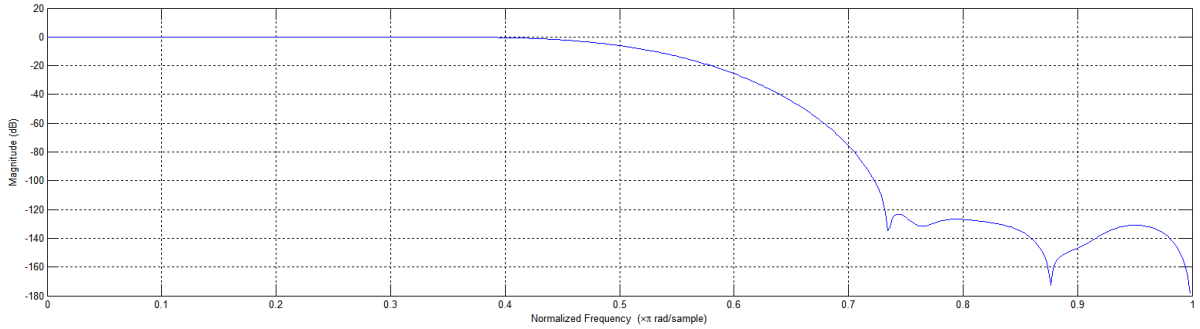


Рисунок 21.35. АЧХ DHB3 фильтра

21.3.1.8.4.4 КИХ фильтры-дециматоры DFIR64

Для максимально гибкого управления окончательной фильтрацией сигнала в канале обработке приемного тракта предусмотрены два включенных последовательно КИХ фильтра с произвольными коэффициентами с порядком до 64 при симметричной или антисимметричной характеристике и до 32 при несимметричной. Порядок фильтра может быть как четный, так и нечетный. Разрядность коэффициентов составляет 24 бита. Коэффициент децимации может принимать значения от 1 до 16.

Для фильтра существует ограничение обрабатываемого сигнала по частоте дискретизации, т.к. выполнение операции фильтрации занимает CEIL(ORDER/2) тактов на каждый отсчет входного сигнала.

Блок имеет два конфигурируемых профиля. Профиль – набор коэффициентов и параметров фильтра (порядок фильтра, коэффициент децимации, масштабирования). Для каждого из профилей предусмотрен отдельный набор регистров управления. В каждый момент времени задействован один из профилей. Профили могут перепрограммироваться и переключаться в ходе работы. Таким образом, можно запрограммировать неактивный профиль во время работы активного и переключиться между ними прямой командой или по сигналу SYNC_SFG. Переключение профилей задерживается автоматически до окончания вычисления очередного отсчета фильтра, что исключает порчу данных при переключении.

Поле PROFILE регистра DFIRx_CFG (x – номер устройства DFIR64, здесь и далее в разделе) отражает при чтении активный в настоящий момент профиль. Запись в поле приводит формированию запроса на переключение. Поле SYNCCFGSENCE включает синхронизацию переключения профилей по сигналу SYNC_CFG.

Коэффициенты фильтра записываются в регистры с наименованием DFIRx_Py_CFk, где y заменяется на номер профиля, k – на номер коэффициента. Остальные параметры профиля управляются полями регистра с наименованием DFIRx_PROFILEy.

При записи коэффициентов в регистры 24-битное значение в 32-битном слове должно быть выровнено по младшему биту. При чтении коэффициентов из регистров они расширяются знаком до 32 бит.

В режимах симметричной и антисимметричной характеристики фильтров в памяти содержится только половина коэффициентов. При использовании нечетного порядка фильтра центральный коэффициент должен быть записан в регистр как ½ своего значения. При порядках фильтра менее предельных коэффициенты загружаются в регистры с младшими номерами, а незадействованные регистры заполнять не обязательно.

Память коэффициентов не определена при включении питания и не сбрасывается по сигналу сброса, поэтому подлежит обязательной инициализации перед включением фильтра в работу.

Поле SYM выбирает вид симметрии характеристики (симметричная при SYM=0, антисимметричная при SYM=1, несимметричная при SYM=2,3).

Поле ORDER задает порядок фильтра. Число отводов фильтра равно ORDER+1.

Поле DR задает коэффициент децимации, который устанавливается равным DR+1.

Поле SCALE задает коэффициент масштабирования, определяя величину сдвига влево значений 55-разрядных аккумуляторов фильтра перед взятием из них старших 26 бит для подачи на выход.

21.3.1.8.4.5 Регулировка амплитуды и фазы выходного сигнала RX_ FGAIN

Основное предназначение устройства – точная регулировка амплитуды и фазы выходного сигнала канала приемного тракта перед подачей на устройство суммирования. Таким образом могут быть реализованы, например, алгоритмы пространственной обработки сигналов.

Операция реализуется комплексным умножением выходного отсчета канала обработки на комплексный весовой коэффициент, компоненты которого задаются в полях разрядностью 16 бит COFF_RE и COFF_IM регистра RX_FGAIN_COFF. Представление компонент коэффициента – двоично-дополнительное. Операция выполняется за 4 такта.

Выходные данные нормируются и приводятся к выходной разрядности таким образом, что коэффициент передачи блока равен единице при COFF_RE=8192 и COFF_IM=0.

Результаты операции проверяются на переполнение и ограничиваются с насыщением при перегрузке. В случае возникновения перегрузки в выходном сигнале, формируется сигнал прерывания.

Управление устройством возможно через поля регистра RX_FGAIN_CTRL.

Собственный блок оценки уровня сигнала BWGAUGE подключен к выходу устройства. Результаты оценки выводятся в поле регистра RX_FGAIN_CTRL.

21.3.1.8.5 Сумматор каналов RX_SUM

RX_SUM предназначен для суммирования квадратурных сигналов до 4х каналов приемного тракта. Блок предназначен для работы в условиях строго синхронной работы каналов. Суммирование сигналов производится в соответствии с заданной маской SUM_MASK. Единица, установленная в маске, указывает, что канал участвует в суммировании. Первому каналу соответствует младший бит в маске. Выходная разрядность сумматора позволяет не допускать переполнения. На выходе сумматора установлен блок масштабирования. В режиме блокирования (SUM_MASK = 4b'0000) блок не выполняет никаких преобразований и не реагирует на сигналы синхронизации. Выходная шина данных в нулевом состоянии.

Значения SCALE и SUM_MASK записываются в регистр управления, но не обязательно применяются сразу. Применение значений может осуществляться по сигналу SYNC_CFG и/или по факту записи коэффициента в регистр и определяется битами в регистре управления.

Если происходит одновременно и запись новых значений, и установка флага UPDATE_WR, то перезапись производится сразу и в рабочий регистр.

Входные сигналы SYNC_CFG сначала маскируются «публичным» значением маски, объединяются по операции «ИЛИ». Объединенный сигнал SYNC_CFG является стробом перезаписи значений из «публичного» в «рабочий» регистр и выдается с задержкой наружу как SYNC_CFG_OUT. Задержка необходима для компенсации задержки в конвейере и сохранения привязки строба к данным

Перезапись осуществляется синхронно в один такт CLK при условии, что блок закончил выполнять операции или не производит вычислений, и перезапись не повлияет на результат вычисления. По окончании перезаписи поле UPDATE_ACK устанавливается в единицу. При записи коэффициента UPDATE_ACK устанавливается в ноль.

Если обновление происходит по сигналу SYNC_CFG, то обновленные значения применяются к тому отсчету входных данных, который был маркирован сигналом синхронизации.

В режиме суммирования каналов блок суммирует каналы в соответствии с маской, хранимой в рабочем регистре. При этом также происходит маскирование рабочим регистром сигналов DV_IN и SYNC_START. Маскированные сигналы объединяются по операции «И». По объединенным сигналам производятся вычисления и формируются выходные сигналы синхронизации и сопровождения с учетом задержки в конвейере вычислений.

При рассогласовании каналов сигналы DV_IN суммируемых каналов могут приходить в разные моменты времени. В этом случае объединенные синхросигналы и сигнал DV обработки будут неактивны и вычисления не смогут быть выполнены. При этом пользователь уведомляется о рассинхронизации каналов прерыванием SYNC_ERR.

21.3.1.8.6 Согласованный фильтр MTCFLT128

Блок предназначен для вычисления свертки комплексного сигнала с комплексными отсчетами заданной импульсной характеристики с целью детектирования появления определенного сигнала на входе. Блок реализован как несимметричный комплексный КИХ фильтр с максимальным порядком 128. Разрядность комплексных отсчетов импульсной характеристики составляет 2x8 бит.

На вход фильтра поступают старшие 8 разрядов сигнала с выхода устройства масштабирования сумматора каналов. Результатом работы являются 4 скалярных произведения разрядностью 32 бита, вычисленные для всех комбинаций между одной из двух компонент сигнала и одной из двух компонент импульсной характеристики. Выходные данные масштабируются до разрядности 16 бит и упаковываются в 64-битное слово. Порядок упаковки приведен в таблице:

OUT64[63:48]	SCALE_OUT_I_RE[15:0]
OUT64[47:32]	SCALE_OUT_I_IM[15:0]
OUT64[31:16]	SCALE_OUT_Q_RE[15:0]
OUT64[15:0]	SCALE_OUT_Q_IM[15:0]

Здесь обозначения «RE» и «IM» относятся к импульсной характеристике, «I» и «Q» - к отсчетам входного сигнала.

Фильтр реализован на 4 вычислительных ядрах, каждое из которых может выполнять свертку на интервале до 32 отсчетов сигнала. Поле ORDER регистра MTCFLT_CFG задает единый порядок для всех ядер. При этом память сигнала и импульсная характеристика разбиваются на 4 фрагмента длиной ORDER+1. Фрагменты сигнала распределяются по ядрам автоматически, в соответствии с установкой ORDER.

Значение ORDER следует выбирать так, чтобы требуемый порядок фильтра был меньше или равен величине 4*(ORDER+1), но максимально близок к ней. В этом случае будет достигаться минимальное время работы ядер в пересчете на входной отсчет.

Фрагменты импульсной характеристики должны быть выделены явно, и каждый из них должен быть записан в регистры MTCFLT_COREx_TAPy своего ядра, номер которого надо подставить вместо литеры x в имени регистров. Запись следует производить в регистры с последовательными номерами y, начиная с первого регистра диапазона (y=0). Регистры с номерами y > ORDER не влияют на результаты вычисления. В том случае, если общий порядок фильтра меньше, чем 4*(ORDER+1), незадействованные регистры последнего ядра (x=3) с номерами y <= ORDER должны быть обнулены.

Далее приведен пример настроек фильтра и записи коэффициентов.

Пусть требуемая длина импульсной характеристики составляет 15. В этом случае наилучшим выбором для порядка фильтрации ядер будет 4, что соответствует ORDER=3. Тогда коэффициенты фильтра k0..k14 следует распределять по регистрам следующим образом:

MTCFLT_CORE0_TAP0[15:0]	{k0_RE[7:0], k0_IM[7:0]}
MTCFLT_CORE0_TAP1[15:0]	{k1_RE[7:0], k1_IM[7:0]}
MTCFLT_CORE0_TAP2[15:0]	{k2_RE[7:0], k2_IM[7:0]}
MTCFLT_CORE0_TAP3[15:0]	{k3_RE[7:0], k3_IM[7:0]}
MTCFLT_CORE1_TAP0[15:0]	{k4_RE[7:0], k4_IM[7:0]}
MTCFLT_CORE1_TAP1[15:0]	{k5_RE[7:0], k5_IM[7:0]}
MTCFLT_CORE1_TAP2[15:0]	{k6_RE[7:0], k6_IM[7:0]}
MTCFLT_CORE1_TAP3[15:0]	{k7_RE[7:0], k7_IM[7:0]}
MTCFLT_CORE2_TAP0[15:0]	{k8_RE[7:0], k8_IM[7:0]}
MTCFLT_CORE2_TAP1[15:0]	{k9_RE[7:0], k9_IM[7:0]}
MTCFLT_CORE2_TAP2[15:0]	{k10_RE[7:0], k10_IM[7:0]}
MTCFLT_CORE2_TAP3[15:0]	{k11_RE[7:0], k11_IM[7:0]}
MTCFLT_CORE3_TAP0[15:0]	{k12_RE[7:0], k12_IM[7:0]}
MTCFLT_CORE3_TAP1[15:0]	{k13_RE[7:0], k13_IM[7:0]}
MTCFLT_CORE3_TAP2[15:0]	{k14_RE[7:0], k14_IM[7:0]}
MTCFLT_CORE3_TAP3[15:0]	0

21.3.1.9 Передающая часть DFE_ALT

Функциональная схема блока приведена на Рисунок 21.36.

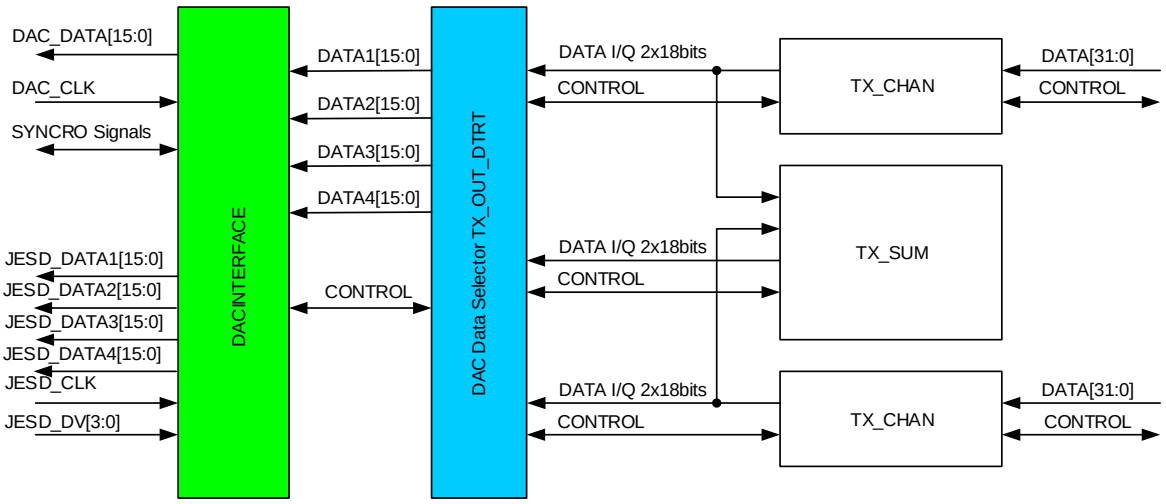


Рисунок 21.36. Блок-схема передающей части DFE

Передающая часть состоит из следующих блоков.

- Интерфейс с ЦАП – DACINTERFACE. Блок предназначен для формирования потока данных для ЦАП в домене тактового сигнала ЦАП в режимах SDR или DDR, JESD204B и режиме, совместимом с JESD207. При этом осуществляется перевод

данных из тактового домена обработки (CLK) в требуемый тактовый домен ЦАП и приведение данных к формату данных ЦАП.

- Коммутатор выходных потоков для ЦАП – TX_OUT_DTRT. Блок предназначен для выбора источника сигнала для ЦАП – одного из TX каналов или весового сумматора.
- Передающий канал цифровой обработки сигнала – TX_CHAN. 2 канала осуществляют модуляцию, канальную фильтрацию, интерполяцию и перенос частоты аналитического сигнала.
- Сумматор каналов TX_SUM выполняет весовое суммирование двух TX_CHAN, работающих в синхронном режиме.

Внутренняя разрядность данных межблочного обмена -18 бит.

21.3.1.9.1 Интерфейс с ЦАП – DACINTERFACE

Блок предназначен для формирования потоков данных для ЦАП по 16-битным параллельным шинам или JESD204B интерфейсу. Поддерживаются следующие варианты интерфейсов.

- Простой параллельный интерфейс
  - SDR
  - DDR с мультиплексированием данных
  - DDR с мультиплексированием каналов
  - DDR с мультиплексированием (уплотнением) отсчетов
- JESD204B
- JESD207
  - SDR
  - DDR
- Оригинальный JESD207-подобный интерфейс трансиверов AD9361

При работе по параллельным интерфейсам в режимах SDR и DDR поддерживается прямо-смещенный и двоично-дополнительный код. В режиме совместимости с JESD207 – только двоично-дополнительный код.

При разрядности ЦАП меньшей, чем разрядность порта, его подключение должно осуществляться к старшим разрядам портов.

Выбор режима работы интерфейса осуществляется через поля регистра DAC_CTRL.

Поле DST_SEL выбирает порт (CMOS, LVDS, JESD204B) или выключает интерфейс при записи «0». Порты CMOS и LVDS обеспечивают передачу потоковых данных в SDR и DDR режимах и могут работать в режиме совместимости с SDR/DDR режимами JESD207.

Поле PBUS_MODE задает режим работы (DDR/SDR/JESD207), и режим мультиплексирования DDR параллельных портов, а поле MODE_OPT1 определяет количество каналов и их разрядность. В зависимости от состояния поля PBUS_MODE, значение поля MODE_OPT1 приводит к установке разных конфигураций.

Режимы работы интерфейса представлены в таблице

Таблица 21.169. Режимы работы DACINTERFACE

	MODE_OPT1	Количество каналов/разрядность	Data Rate	Frame Sync
Simple Parallel SDR PBUS_MODE=0	0	1Ch X 16bit	DAC CLK	-
	1	2Ch X 8bit	DAC CLK	-
Simple Parallel DDR мультиплексирование каналов PBUS_MODE=1	0	2Ch X 16bit	DAC CLK	-
	1	4Ch X 8bit	DAC CLK	-
Simple Parallel DDR мультиплексирование бит PBUS_MODE=2	NA	2Ch X 16bit	DAC CLK	-
Simple Parallel DDR мультиплексирование отсчетов PBUS_MODE=3	0	1Ch X 16bit	2xDAC CLK	-
	1	2Ch X 8bit	2xDAC CLK	-
JESD207 SDR (AD9361 CMOS) PBUS_MODE=4	0	2Ch X 15bit	DAC CLK/2	Есть (LSB)
	1	4Ch X 15bit	DAC CLK/4	Есть (LSB)
JESD207 DDR (AD9361 CMOS) PBUS_MODE=5	0	2Ch X 15bit	DAC CLK	Есть (LSB)
	1	4Ch X 15bit	DAC CLK/2	Есть (LSB)
AD9361 DDR LVDS (DDR) PBUS_MODE=6	0	2Ch X 15bit	DAC CLK	Есть (LSB)
	1	4Ch X 15bit	DAC CLK/2	Есть (LSB)

Поле TC2OB_ON включает режим преобразования двоично-дополнительного кода в прямо-смещенный, которое производится инверсией старшего бита.

При необходимости, фаза тактового сигнала ЦАП может быть точной настройке через поля регистра CLK_SYNC_CFG. Данная процедура описана в Разделе «Точная подстройка задержки тактового сигнала АЦП и ЦАП».

Функциональная блок-схема формирователя потока для CMOS и LVDS интерфейсов представлена на Рисунок 21.37.

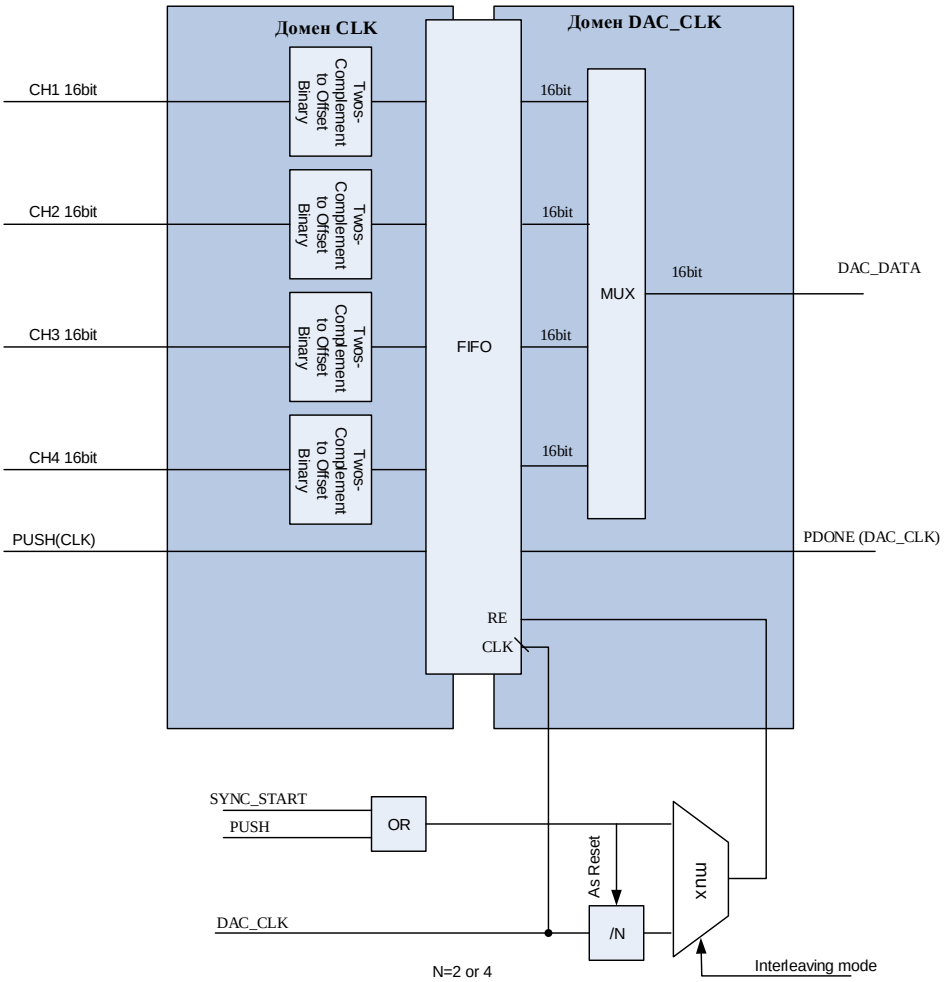


Рисунок 21.37. Блок-схема обработки в LVDS и CMOS интерфейсах ЦАП

FIFO на стыке блоков необходимы для корректного перехода сигнала в другой тактовый домен.

21.3.1.9.1.1 Формирование данных в CMOS и LVDS портах ЦАП в режиме простой SDR параллельной шины данных.

Данные передаются по возрастающему фронту тактового сигнала ЦАП с темпом один отсчет на период. Данные могут интерпретироваться интерфейсом как

- один канал с разрядностью до 16 бит;
- два канала с разрядностью до 8 бит.

Таблица 21.170. Упаковка данных в SDR параллельную шину данных ЦАП

MODE_OPT1=0 (16 bit)	**MODE_OPT1=1 (8 bit)**
DAC_DATA=CH1[15:0]	DAC_DATA[15:8]=CH1[15:8] DAC_DATA[7:0]=CH2[15:8]

21.3.1.9.1.2 Формирование данных в CMOS и LVDS портах ЦАП в режиме простой DDR параллельной шины данных.

Данные передаются по обоим фронтам тактового сигнала с темпом один отсчет на период тактового сигнала. При этом существует следующие варианты передачи данных:

- На шине мультиплексируются (чередуются) биты данных ЦАП;
- На шине мультиплексируются каналы ЦАП.
- На шине чередуются отсчеты канала

Данные в режиме DDR с мультиплексированием бит всегда интерпретируются как два канала по 16 бит. При этом 16 разрядная шина данных интерпретируется как две шины по 8 бит. По старшим 8 битам шины передаются данные первого канала в режиме DDR, по младшим 8 бит – DDR данные второго канала.

При мультиплексировании бит на шине существуют два варианта мультиплексирования:

- в отсчете первым передается младший бит из мультиплексируемой пары разрядов;
- в отсчете первым передается старший бит.

На рисунке ниже приведен пример мультиплексирования битов данных, когда в отсчете передается сначала младший бит. Мультиплексирование бит осуществляется в паре четный/нечетный бит: 15/14, 13/12, …. 1/0.

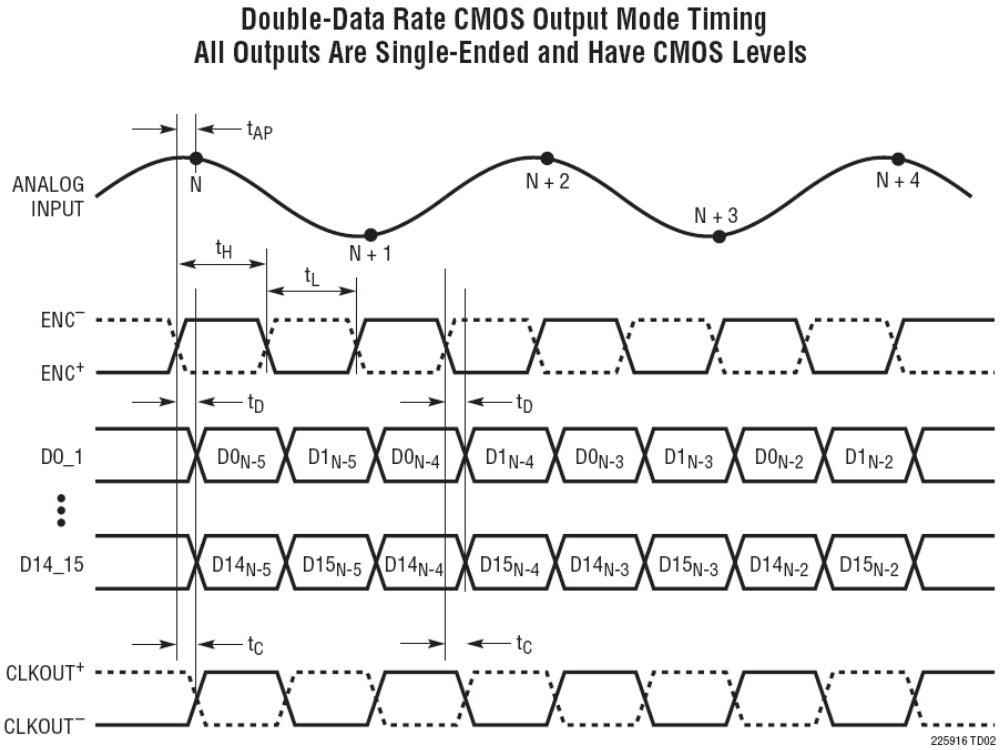


Рисунок 21.38. Временная диаграмма примера мультиплексирования бит в DDR режиме интерфейса ЦАП

Данные в режиме DDR с мультиплексированием каналов могут интерпретироваться двумя способами.

- Два канала с разрядностью до 16 бит.

- Четыре канала с разрядностью до 8 бит. Биты 15:8 соответствуют паре каналов CH1 и CH2, биты 7:0 соответствуют другой паре каналов CH3 и CH4.

Активным фронтом тактового сигнала при передаче данных является возрастающий фронт. В режиме DDR отсчет начинается с возрастающего фронта.

Данные в режиме DDR с чередованием отсчетов могут интерпретироваться двумя способами.

- Один канал CH1 с разрядностью до 16 бит.
- Два канала с разрядностью до 8 бит. Биты 15:8 соответствуют каналу CH1, биты 7:0 соответствуют каналу CH2.

Упаковка данных в параллельную шину ЦАП в режиме DDR определяется следующими таблицами.

Таблица 21.171. Упаковка данных в DDR параллельную шину данных ЦАП с мультиплексированием каналов

MODE_OPT1	Rising Edge	Falling Edge
0	CH1	CH2
1	CH1[15:8] & CH3[15:8]	CH2[15:8] & CH4[15:8]

Таблица 21.172. Упаковка данных в DDR параллельную шину данных ЦАП с мультиплексированием бит данных

MODE_OPT1	Rising Edge	Falling Edge
“0”- Первым следуют младшие (четные) биты.	DAC_DATA[15:8]=CH1[15:0] четные (младшие) DAC_DATA[7:0]= CH2[15:0] четные (младшие)	DAC_DATA[15:8]=CH1[15:0] нечетные (старшие) DAC_DATA[7:0]= CH2[15:0] нечетные (старшие)
“1”- Первым следуют старшие (нечетные) биты.	DAC_DATA[15:8]=CH1[15:0] нечетные (старшие) DAC_DATA[7:0]= CH2[15:0] нечетные (старшие)	DAC_DATA[15:8]=CH1[15:0] четные (младшие) DAC_DATA[7:0]= CH2[15:0] четные (младшие)

Таблица 21.173. Упаковка данных в DDR параллельную шину данных ЦАП с мультиплексированием отсчетов

MODE_OPT1	Rising Edge	Falling Edge
0	$CH1_i$	$CH1_{i+1}$
1	$CH1_i$ [15:8] & $CH2_i$ [15:8]	$CH1_{i+1}$ [15:8] & $CH2_{i+1}$[15:8]

21.3.1.9.1.3 Формирование данных в CMOS шине ЦАП в режиме JESD207 (AD9364 CMOS Full Duplex Dual Port Compatible)

Этот режим предназначен для вывода цифровых данных из DFE по JESD207 интерфейсу. Этот интерфейс совместим с LVCMOS интерфейсом AD9361 в режиме “Full Duplex Dual Port”. При этом возможен как SDR, так и DDR режим. Количество каналов – два или четыре (один или два квадратурных.). Максимальная разрядность данных – 15 бит. Младший нулевой бит шины данных задействован под строб TX_FRAME. При меньшей разрядности

шина данных ЦАП должна подключаться к старшим разрядам шины с выравниванием по MSB. При помощи поля FRAME_AL регистра DAC_CTRL устанавливается активный уровень сигнала фреймовой синхронизации TX_FRAME.

Поскольку интерфейс подразумевает передачу данных только в двоично-дополнительном коде, преобразование формата данных при приеме не производится.

В интерфейсе принято следующее соответствие квадратурных каналов физическим

Канал интерфейса JESD207	Физический канал
CH1_I	CH1
CH1_Q	CH2
CH2_I	CH3
CH2_Q	CH4

21.3.1.9.1.4 Формирование данных в LVDS шине ЦАП в режиме JESD207 (AD9364 LVDS Full Duplex Dual Port Compatible)

Этот режим предназначен для вывода цифровых данных в RFFE типа AD9361 в режиме LVDS “Full Duplex Dual Port”. При этом возможен как SDR, так и DDR режим. Количество каналов – один или два квадратурных. Максимальная разрядность данных – 16 бит (разрядность шины данных 8 бит). Младший нулевой бит шины данных задействован под строб RX_FRAME. При меньшей разрядности данных выходная шина ЦАП должна подключаться к старшим разрядам шины с выравниванием по MSB. При помощи поля OUT_DATA_BW осуществляется маскирование нулем незадействованных младших разрядов.

Поскольку интерфейс подразумевает передачу данных только в двоично-дополнительном коде, преобразование формата данных при приеме не производится.

Режим отличается тем, что отсчеты данных передаются половинками по шине вдвое меньшей разрядности, чем разрядность данных.

Пример подключения 12 битного ЦАП к интерфейсу представлена на Рисунок 21.39.

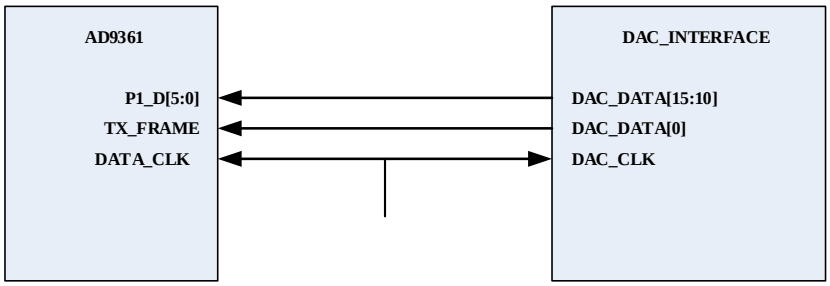


Рисунок 21.39. Пример подключения AD9361 в режиме LVDS “Full Duplex Dual Port”

Ниже на рисунках представлены временные диаграммы интерфейса для SDR и DDR передачи данных одного и двух каналов.

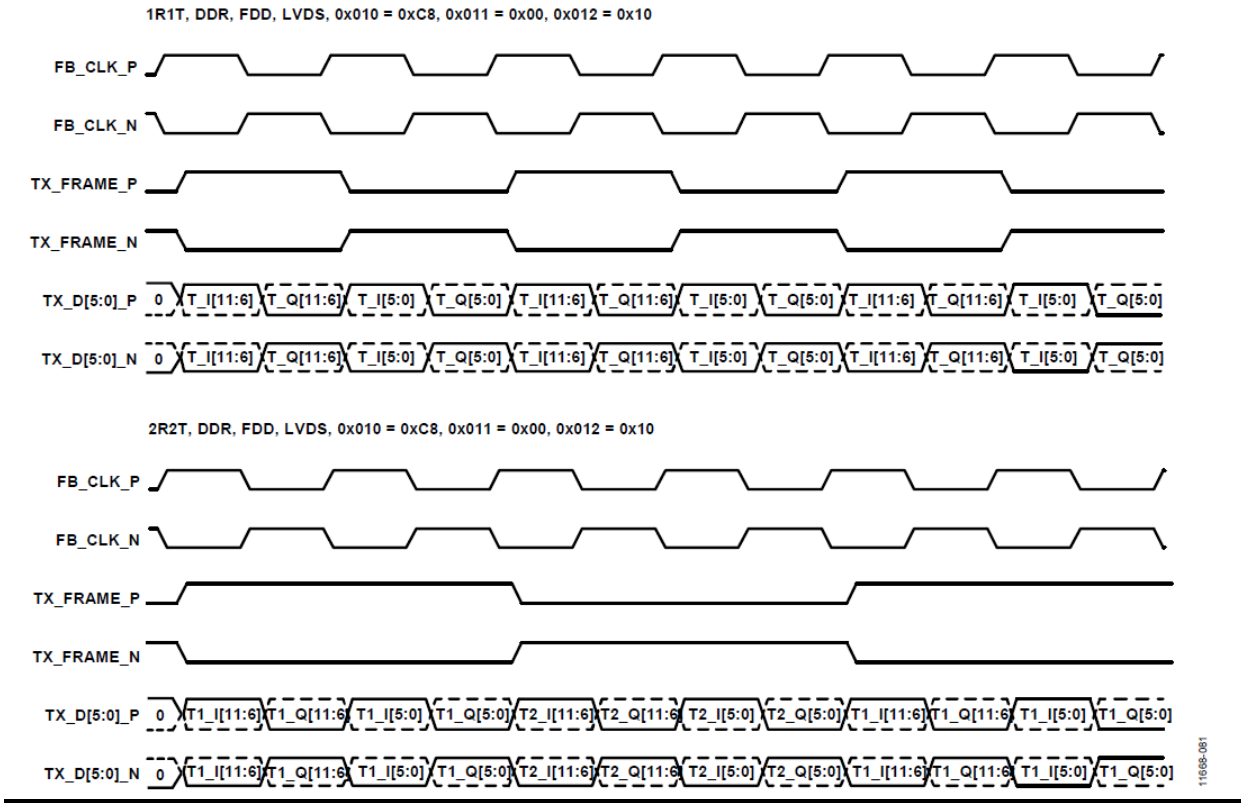


Рисунок 21.40. Временные диаграммы интерфейса AD9361 в режиме LVDS “ Full Duplex Dual Port”

На диаграмме TX_D[5:0] соответствует старшим шести битам шины данных ЦАП DAC_DATA[15:10].
В интерфейсе принято следующее соответствие квадратурных каналов физическим

Канал интерфейса JESD207	Физический канал
T_I, T1_I	CH1
T_Q, T1_Q	CH2
T2_I	CH3
T2_Q	CH4

Актуальная разрядность шины данных ЦАП должна быть задана MODE_OPT2 регистра DAC_CTRL. Это является необходимым для правильной сборки половинок отсчета, последовательно передаваемых по шине, в целые отсчеты. Сигнал фреймовой синхронизации принимается по нулевому разряду шины данных ЦАП и сигнализирует начало пачки из 4 отсчетов, принимаемых по шине, соответствующей одному комплексному отсчету, компоненты которого разбиты на половинки. По окончанию приема всех 4 частей формируется комплексный отсчет сигнала полной разрядности.

Интерфейс подразумевает передачу только квадратурных данных (I и Q компоненты). Количество каналов – 1 или 2.

21.3.1.9.1.5 Формирование данных в режиме JESD204

Данные каналов CH1..CH4 передаются без всяких дополнительных преобразований.

Четыре канала JESD204 интерфейса соответствуют четырем входным сигналам CH1…CH4 DACINTERFACE.

21.3.1.9.1.6 Требование к тактовым частотам интерфейса.

Для корректной работы блока требуется, чтобы тактовая частота CLK была не ниже тактовой частоты ЦАП или частоты следования отсчетов в JESD204. В противном случае произойдет опустошение FIFO и блок сформирует сигнал CLK_ERR. CLK_ERR отображается в одноименное поле регистра DAC_CTRL и сигнализирует о событии чтения пустого FIFO. Данная ситуация является аварийной и требует устранения причины (перенастройки частот) с последующим перезапуском передатчика.

21.3.1.9.1.7 Процедура PUSH.

Процедура PUSH предназначена для предварительного заполнения трактов обработки и FIFO DACINTERFACE данными вплоть до выходного порта ЦАП валидными данными. Процедура позволяет начать передачу валидных данных на ЦАП сразу после старта.

По команде PUSH (запись единицы в поле PUSH) имитируется сигнал внешней синхронизации старта, что разрешает работу блоков как в обычном режиме. После подачи команды необходимо начать загрузку данных через DMA. До появления данных на выходе DMA установлены нулевые значения. Вместе с началом поступления валидных данных из канала DMA поднимается сигнал PUSH_MARK и начинает передаваться по цепочке блоков вместе с первым отсчетом данных. Во всех устройствах передающего тракта, включая интерфейс ЦАП, предусмотрен механизм синхронизации времени прохождения сигнала PUSH_MARK с отсчетом обрабатываемого сигнала. Таким образом, появление сигнала PUSH_MARK на выходе интерфейса ЦАП означает готовность к выдаче первого валидного отсчета на шину данных ЦАП. Как только это произошло, процедура PUSH считается законченной и чтение данных из DMA приостанавливается, замораживая состояние всех устройств до момента установки сигнала SYNC_START. На протяжении всей процедуры PUSH и времени ожидания SYNC_START на шине ЦАП установлены нулевые значения.

В момент установки SYNC_START сразу же выдается первый отсчет на шину ЦАП и разрешается чтение с выхода DMA. Таким образом обеспечивается синхронный, без задержек старт передачи.

Состояние процедуры PUSH отражается в поле PUSH. Ноль в поле означает, что процедура не стартовала или еще выполняется. Единица означает, что процедура выполнилась полностью и передающий тракт находится в состоянии ожидания внешнего сигнала старта.

По окончании процедуры PUSH вырабатывается сигнал прерывания TX_PUSH_DONE.

Если сигнал SYNC_START=1 принимается до окончания процедуры PUSH, то продолжается нормальная работа интерфейса.

В случае режима JESD207 привязка начала потока осуществляется к ближайшему положению сформированного во время процедуры PUSH внутреннего сигнала TX_FRAME.

21.3.1.9.2 Коммутатор выходных потоков передающей части TX_OUT_DTRT

Блок предназначен для коммутации потоков данных в передающей части DFE_ALT. В нем осуществляется выбор источников сигнала для четырех каналов интерфейса ЦАП, выбирается сигнал для записи с помощью PROBE, осуществляется оценка параметров передаваемых сигналов и коррекция смещения нуля в них.

Блок схема коммутации для одного из каналов интерфейса ЦАП приведена на Рисунок 21.41.

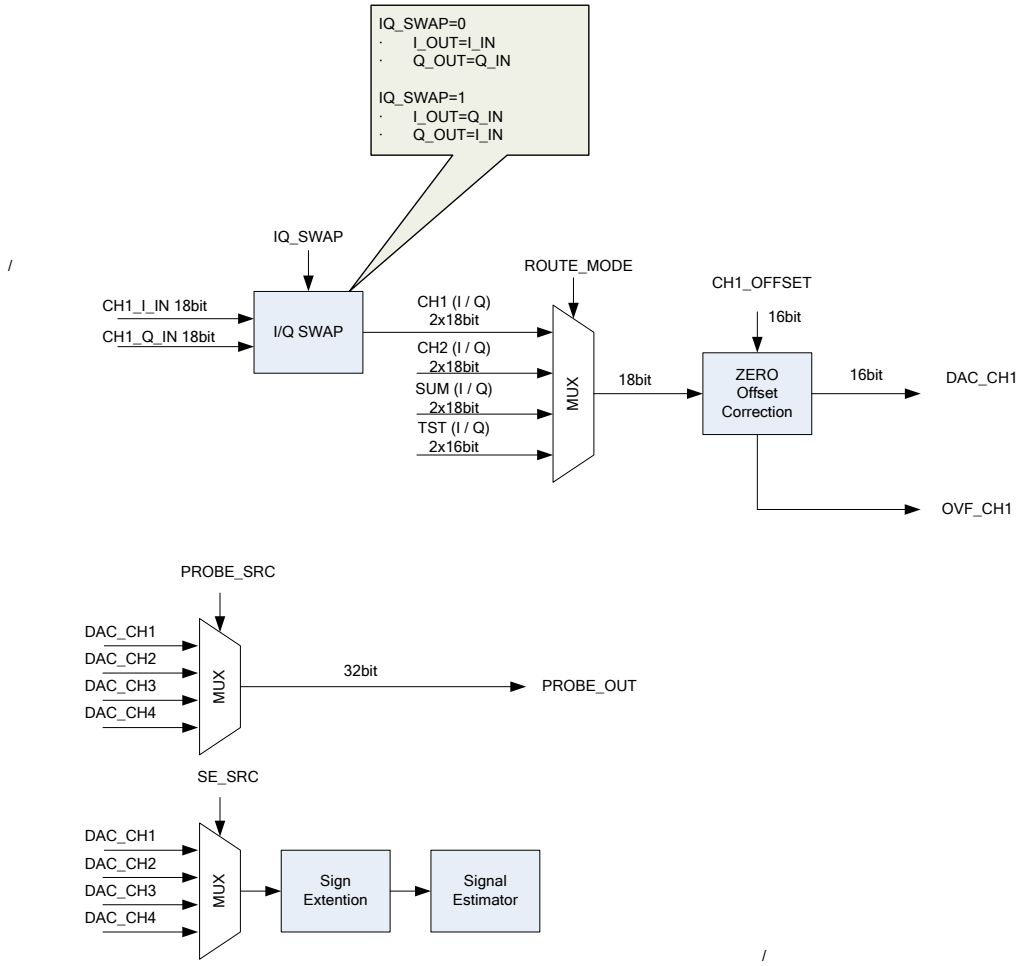


Рисунок 21.41. Блок-схема селекции для формирования выходных сигналов коммутатора

Сигналы Q_SWAP и ROUTE_MODE соответствуют одноименным полям регистра TX_OUT_DTRT_ROUTE_CTRL. Коммутация выполняется в соответствии с таблицей.

Таблица 21.174. Выбор режима коммутации TX_OUT_DTRT

ROUTE_MODE	DAC_CH1	DAC_CH2	DAC_CH3	DAC_CH4

0	CH1_I_IN	CH1_Q_IN	CH2_I_IN	CH2_Q_IN
1	CH1_I_IN	CH2_I_IN	CH1_Q_IN	CH2_Q_IN
2	SUM_I_IN	SUM_Q_IN	0	0
3	TST1	TST2	TST1	TST2

Выбор источников для подачи на PROBE и SE (Signal Estimator) задается полями PROBE_SRC и SE_SRC регистра TX_OUT_DTRT_TST_CTRL.

21.3.1.9.2.1 Настройка коррекции смещения нуля

Коррекция смещения нуля в каналах осуществляется прибавлением константы к входным отсчетам. К 18-разрядному отсчету прибавляется значение 18 бит, полученных расширением знака из младших 16 разрядов регистра TX_OUT_DTRT_CHx_OFFSET. Операция суммирования производится с контролем переполнения и ограничением с насыщением. На выход подаются старшие 16 разрядов.

Для подбора оптимального значения параметра коррекции может быть использован блок оценки параметров сигнала.

21.3.1.9.2.2 Блок оценки параметров сигнала

Блок оценки параметров сигнала (Signal Estimator) встроен в коммутатор и имеет возможности для оценки следующих параметров сигнала:

- мощность;
- максимальное и минимальное значение;
- среднее значение сигнала.

Блок-схема устройства совпадает со схемой такого же блока оценки, установленного в селекторе приемных каналов.

Оценка параметров сигнала производится на длительности выборки, задаваемой через поле SE_EST_DURATION регистра TX_OUT_DTRT_TST_CTRL. Значение поля интерпретируется как степень числа 2, т.е. длительность выборки составляет $2^{SE\_EST\_DURATION}$ отсчетов.

Для оценки среднего значения (постоянной составляющей) и мощности сигнала производится деление его на кол-во накопленных отсчетов, реализованное как сдвиг вправо на величину SE_EST_DURATION.

Оценка мощности сигнала выводится в формате с плавающей точкой через поля MANT и EXP регистра TX_OUT_DTRT_SE_POWRMS. Разрядности как мантиссы, так и экспоненты составляют 8 бит.

Минимум, максимум и оценка среднего значения могут быть прочитаны в младших 16 битах регистров, соответственно, TX_OUT_DTRT_SE_MIN, TX_OUT_DTRT_SE_MAX и TX_OUT_DTRT_SE_MEAN.

Сигнал старта блока оценки сигнала формируется по факту записи единицы в поле SE_START_STATUS регистра TX_OUT_DTRT_TST_CTRL. Если предыдущая процедура измерения не закончена, она прерывается.

По событию окончания измерений формируется сигнал прерывания INT_SE и устанавливается в «1» сигнал готовности, значение которого можно узнать, прочитав поле SE_START_STATUS регистра TX_OUT_DTRT_TST_CTRL.

21.3.1.9.2.3 Механизм выравнивания потоков данных.

В TX тракте все выходные потоки данных от каналов и сумматора имеют одну и ту же скорость (темп поступления данных). При этом скорость входных потоков каналов может существенно различаться из-за того, что в каналах могут быть различные коэффициенты интерполяции. Также в каналах может быть различная задержка в обработке потоковых данных по причинам разных порядков фильтра и различия в стадиях обработки. Поэтому от разных каналов, отличающихся между собой конфигурацией, начала потоков, сопровождаемых маркерами PUSH, могут прийти на вход сумматора и/или роутера в различное время.

Для выравнивания потоков по времени (выравнивание начала потоков) предназначен специальный механизм. В зависимости от режима ROUTE_MODE, механизм будет отличаться.

Режимы 0 и 1. Режимы предназначены для подключения непосредственно выходов каналов. Коммутация в режиме включенного и выключенного выравнивания представлена на Рисунок 21.42 и Рисунок 21.43.

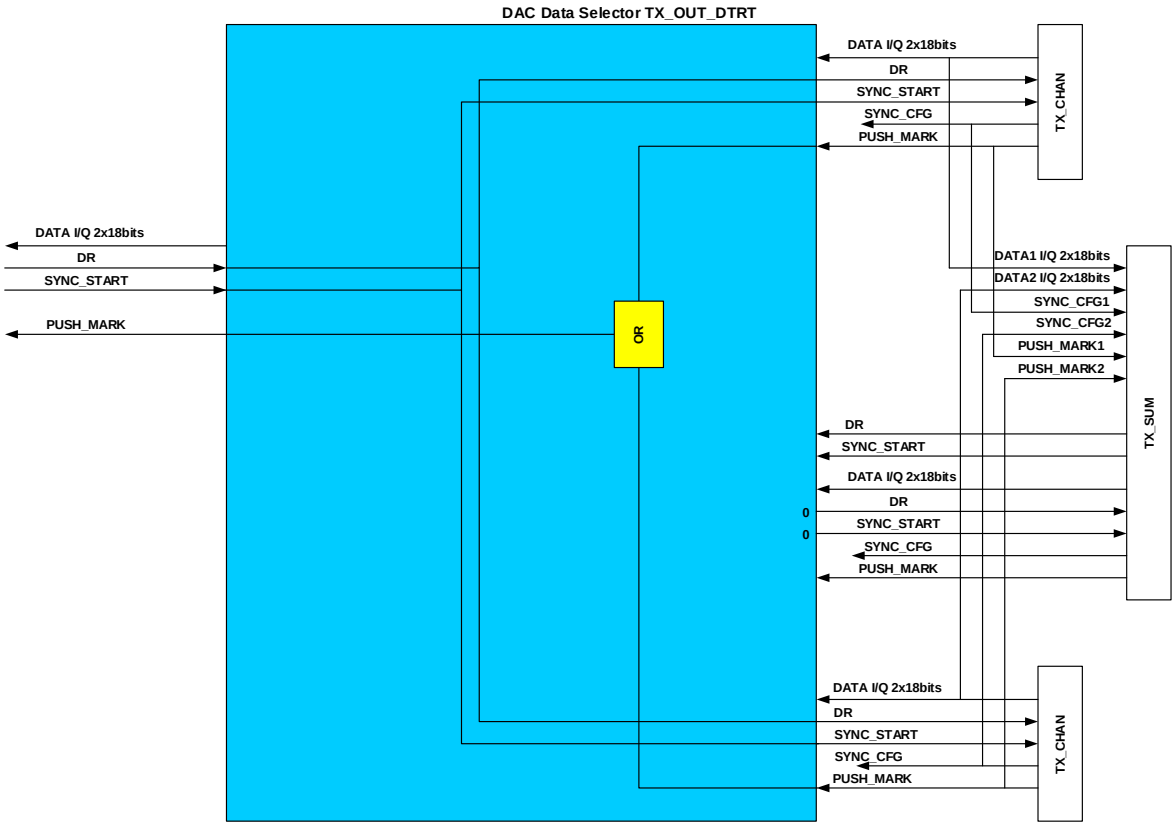


Рисунок 21.42. Схема выключенного выравнивание TX_OUT_DTRT в ROUTE_MODE =0,1

Сигналы синхронизации старта и DR разводятся на два канала. Выходной сигнал PUSH_MARK образуется операцией «ИЛИ» из выходных PUSH_MARK каналов. В этом случае отключение одного из каналов не приводит к «зависанию» процедуры PUSH. Самый «ранний» поток определяет окончание процедуры PUSH.

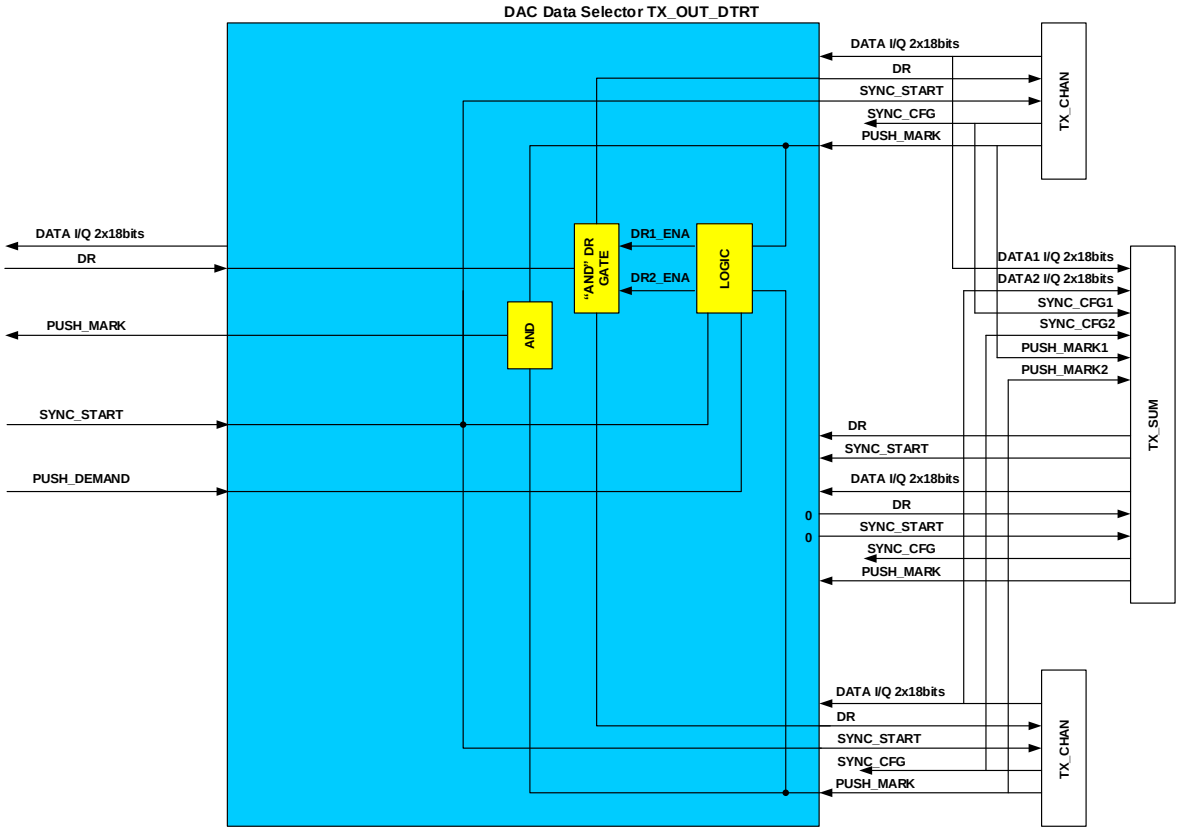


Рисунок 21.43. Схема включенного выравнивание TX_OUT_DTRT в ROUTE_MODE =0,1

Сигналы синхронизации старта разводится на два канала. По окончании процедуры PUSH процедура останавливается падением SYNC_START от DAC интерфейса. Выравнивание потоков осуществляется приостановкой формирования сигнала DR для того канала, от которого сигнал PUSH_MARK пришел первым. Выходной сигнал роутера PUSH_MARK получается объединением по операции «И» выходных PUSH_MARK каналов. Тем самым обеспечивается выравнивание потоков по их началу на входе DAC интерфейса.

Режим 2. Режим предназначены для подключения к выходу сумматора каналов. Коммутация в режиме включенного и выключенного выравнивания представлена на Рисунок 21.44 и Рисунок 21.45.

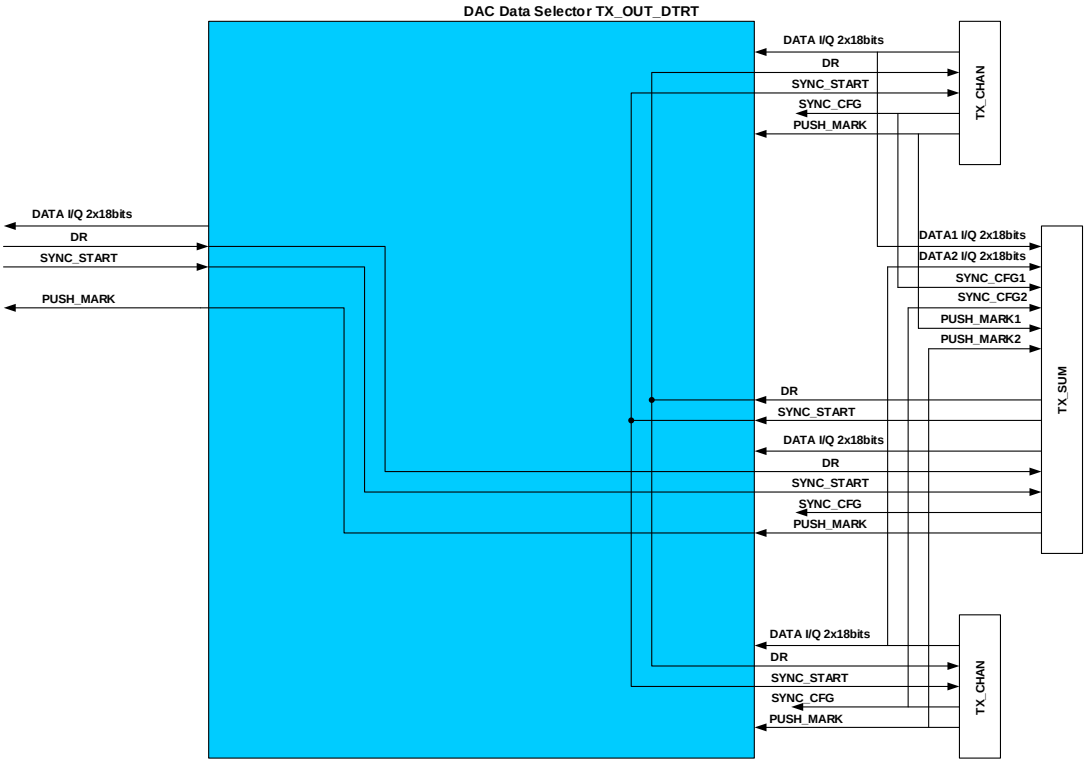


Рисунок 21.44. Схема выключенного выравнивание TX_OUT_DTRT в ROUTE_MODE =2

Выходной сигнал PUSH_MARK сумматора каналов образуется операцией “ИЛИ” из выходных PUSH_MARK каналов внутри сумматора.

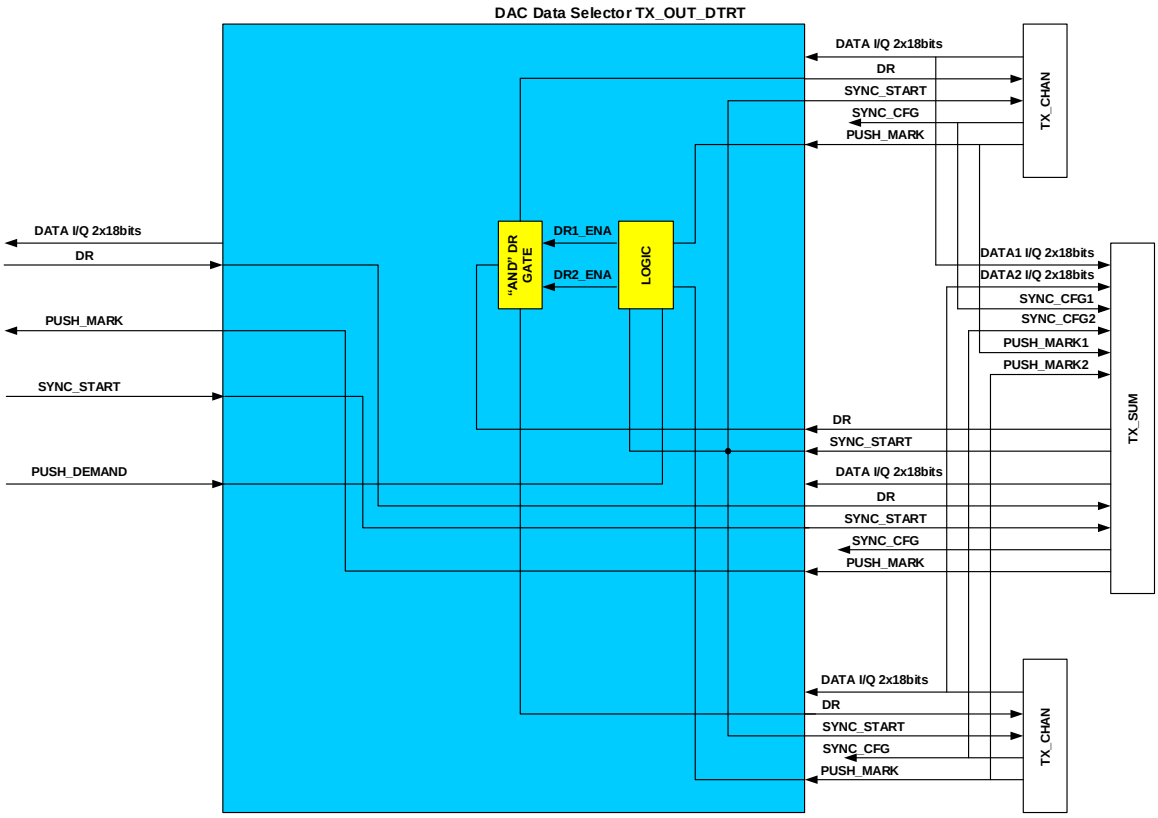


Рисунок 21.45. Схема включенного выравнивание TX_OUT_DTRT в ROUTE_MODE =2

Коммутация похожа на коммутацию в режимах ROUTE_MODE 0 и 1 и отличаются использованием задающих сигналов DR, SYNC_START. Сигналы DR для каналов образуются из выходного сигнала DR сумматора. Аналогичным способом осуществляется маскирование сигнала DR для «опережающего» канала. Этот механизм должен обеспечить выравнивание начал потоков на входах сумматора вместе с сопровождающими их сигналами PUSH_MARK. Выходной сигнал сумматора PUSH_MARK оказывается автоматически выровненным с началом обоих потоков каналов и коммутируется на вход DAC интерфейса.

21.3.1.9.3 Сумматор каналов TX_SUM

TX_SUM предназначен для суммирования квадратурных сигналов двух каналов передающего тракта. Блок предназначен для работы в условиях строго синхронной работы двух каналов. На выход передаются старшие 18 бит суммы, что соответствует коэффициенту усиления 0.5 по каждому каналу. Управление сумматором доступно через поля регистра TX_SUM_CTRL.

Суммирование производится с учетом маски, которая задается полем SUM_MASK. Бит SUM_MASK[1] отвечает за первый канал, SUM_MASK[0] – за второй. Единица в бите маски означает, что канал участвует в суммировании.

Регистр TX_SUM_CTRL разрешен на запись при любом значении SYNC_START. Значение поля SUM_MASK прозрачно (всегда переписывается в рабочий регистр) при неактивном SYNC_START.

Перезапись значение SUM_MASK из регистрового файла во внутренние регистры блока при активном SYNC_START может осуществляться по сигналу SYNC_CFG и/или по событию записи коэффициента в регистр. Это определяется битами в регистре TX_SUM_CTRL.

- UPDATE_WR=1 разрешает перезапись в рабочий регистр каждый раз при записи в регистр управления.
- UPDATE_SYNC=1 разрешает перезапись в рабочий регистр каждый раз при приходе любого из синхросигналов SYNC_CFG.

Перезапись осуществляется синхронно в один такт CLK при условии, что блок закончил выполнять операции и не производит вычислений. При этом перезапись не повлияет на результат текущего вычисления. По окончании перезаписи поле UPDATE_ACK устанавливается в единицу. При записи регистра UPDATE_ACK устанавливается в ноль.

Если обновление происходит по сигналу SYNC_CFG, то обновленные значения применяются к тому отсчету входных данных, который был маркирован сигналом синхронизации.

21.3.1.9.4 Передающий канал цифровой обработки сигнала - TX_CHNL

Передающий тракт DFE_ALT включает в себя 2 канала обработки сигнала. Каждый канал осуществляет модуляцию, фильтрацию, интерполяцию и перенос частоты одного цифрового квадратурного сигнала.

Каждый канал управляется регистрами, доступными при включении нужной страницы в диапазоне адресов 0x12000-0x12FFF. Выбор режима адресации и страницы осуществляется через регистр TX_CTRL. Доступен групповой режим адресации, при котором управляющие данные записываются одновременно в оба канала, но чтение данных невозможно.

Общее управление каналом доступно через регистр TX_CHNL_CTRL.

Битом ENABLE можно включить (1) и выключить (0) канал.

Бит STATE отражает состояние синхронизации: «0» – ожидание SYNC_START, «1» – прием данных и обработка.

Бит SYNC_EVENT устанавливается при достижении сигналом SYNC_CFG выхода канала.

Бит PPS_EVENT устанавливается при возникновении события PPS.

Поле PM_OUT при чтении сигнализирует (1) об окончании процедуры PUSH.

Поле OVF_STATUS является 8-битным составным индикатором перегрузок. Каждый его бит отражает статус перегрузки определенного устройства (см. описание регистра).

Биты SYNC_EVENT, PPS_EVENT, PM_OUT и биты поля OVF_STATUS сбрасываются записью в них значения «1».

Поле MODE управляет включением (1) режима работы под управлением планировщика задач. Канал может работать как без планировщика заданий в ручном режиме управления, так и под управлением встроенного планировщика. Управление синхронизацией передатчика рассматривается в Разделе 21.3.1.2.

Поле INVSINC_ENA разрешает (1) работу INVSINC фильтра, установленного на выходе блока гетеродина.

Обобщенная блок-схема MC03DFE_TX_CHNL приведена на Рисунок 21.46.

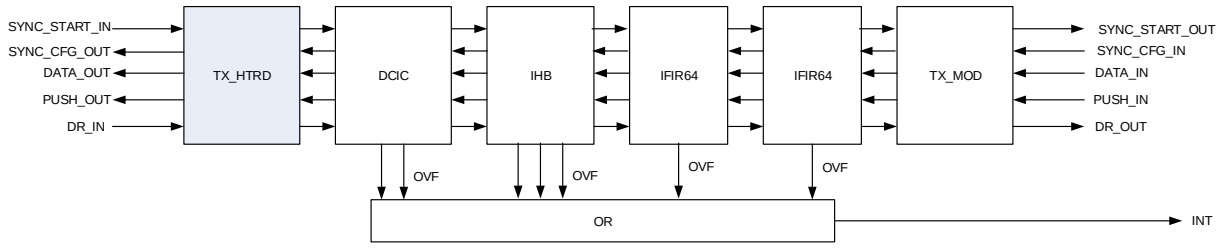


Рисунок 21.46. Блок-схема передающего канала TX_CHNL

Разрядность данных межблочных соединений 18 бит.

21.3.1.9.4.1 Модулятор TX_MOD

Модулятор осуществляет распаковку входных данных, преобразование отсчетов по заданным в таблице законам и прямое расширение спектра сигнала PN последовательностью. Максимальная длина PN последовательности составляет 256 чипов. Блок имеет 16 профилей (набор из 16 различных последовательностей) с возможностью их оперативного переключения. На выходе блока формируется квадратурный сигнал.

Блок обеспечивает следующие режимы, выбираемые полем MODE регистра MOD_MODE:

MODE=0: прямая передача данных в квадратуры;

MODE=1: преобразование по таблице (MEMORY_MAP, QAM);

MODE=2: прямое расширение спектра, символьный режим;

MODE=3: прямое расширение спектра, битовый режим;

Интерпретация формата упакованных входных данных зависит от установленного режима работы.

21.3.1.9.4.1.1 Режим работы DIRECT_DATA, MODE=0

Входное 32 битное слово интерпретируется как один отсчет квадратурного сигнала разрядностью 2x16 бит в двоично-дополнительно кодировке.

{I[15..0], Q[15..0]} = MOD_INP[31..0]

Комплексный отсчет без преобразований поступает на выход блока. Частота дискретизации не меняется.

21.3.1.9.4.1.2 Режим работы MEMORY MAP (QAM), MODE=1

В 32 битном слове входных данных упакованы 4 символа разрядностью по 8 бит.

{D4[7..0], D3[7..0], D2[7..0], D1[7..0]} = MOD_INP[31..0]

Данная операция обычно производится при QAM модуляции. В процессе преобразования каждому символу становится в соответствие комплексный вектор разрядностью 2x8 бит, выбираемый из таблицы по значению символа. Таблица расположена в собственной памяти модулятора, доступной через регистры MOD_MEMx. При загрузке массива векторов в таблицу литера x в имени регистра соответствует содержимому символа D, для которого загружается вектор. Всего доступно 256 векторов, что соответствует разрядности символов 8 бит.

Частота дискретизации сигнала на выходе блока равна частоте на входе.

Значение компонент вектора передается в старших битах компонент выходного сигнала, младшие 8 бит обнуляются.

21.3.1.9.4.1.3 Прямое расширение спектра, символьный режим, MODE=2

В 32 битном слове входных данных упакованы 4 комплексных символа разрядностью 2x4 бита.

{I4[3..0], Q4[3..0], I3[3..0], Q3[3..0], I2[3..0], Q2[3..0], I1[3..0], Q1[3..0]} = MOD_INP[31..0]

Символы не являются представлением комплексных отсчетов сигнала. Каждая компонента интерпретируется как индекс выбора из набора, загруженного в память модулятора, расширяющей последовательности для соответствующей выходной компоненты. Всего в наборе 16 последовательностей длиной до 256 чипов. Переключая выбираемые последовательности с помощью символов, можно составлять более длинные последовательности, вплоть до 256*16=4096 отсчетов.

Последовательности должны загружаться в память модулятора частями по 16 бит через регистры MOD_MEMx. На каждую последовательность отведен блок из 16 регистров, начинающийся с регистра, имеющего номер, кратный 16. Номер регистра x, с которого начинается нужный диапазон для загрузки каждой последовательности, определяется как x = n*16, где n соответствует значению компоненты символа, выбирающего данную последовательность. Загружаемая последовательность будет передаваться, начиная с

младшего бита слова, загруженного в первый регистр диапазона. При значении чипа последовательности «1» на выход передается 127, при «0» передается -127. Значение передается в старших битах компонент выходного сигнала, младшие 8 бит обнуляются.

Длина последовательности определяется значением поля SEQ_LENGTH регистра MOD_PN_PROFILE. Используются чипы последовательности, начиная с нулевого и до номера SEQ_LENGTH-1. После передачи SEQ_LENGTH чипов из распаковщика запрашивается новый символ. Частота дискретизации сигнала на выходе в SEQ_LENGTH раз больше, чем на входе.

21.3.1.9.4.1.4 Прямое расширение спектра, битовый режим, MODE=3

Входное 32 битное слово интерпретируется как 16 упакованных отсчетов квадратурного сигнала разрядностью 2x1 бит.

{I16, Q16, I15, Q15 … I2, Q2, I1, Q1} = MOD_INP[31..0]

Номера расширяющих последовательностей задаются полями I_PROFILE, Q_PROFILE регистра MOD_PN_PROFILE. Загрузка последовательностей в память производится так же, как в режиме MODE=2.

Для каждой компоненты каждого отсчета входного сигнала передается часть выбранной последовательности длиной, определяемой в поле SEQ_LENGTH. Каждый чип последовательности подвергается операции «исключающее ИЛИ» с текущим значением компоненты входного отсчета. На соответствующий компоненте выход модулятора подается -127 при результате операции, равном 1 и 127 в противном случае. Значение передается в старших битах компонент выходного сигнала, младшие 8 бит обнуляются.

21.3.1.9.4.2 КИХ фильтры-интерполяторы IFIR64

В канале обработки передающего тракта имеется 2 включенных последовательно фильтра-интерполятора с программируемыми коэффициентами. Фильтры могут использоваться для коррекции АЧХ передающего тракта трансивера и канальной фильтрации. Разрядность коэффициентов фильтров – 24 бита. Порядок фильтра может быть как четный, так и нечетный. При симметричной или антисимметричной импульсной характеристики фильтра достигается максимальный порядок в 64 отсчета. При отсутствии симметрии максимальный порядок составляет 32.

Интерполяция входных отсчетов данных осуществляется перед фильтрацией либо повторением отсчетов, либо добавлением нулевых отсчетов.

Для фильтра существует ограничение обрабатываемого сигнала по частоте дискретизации, т.к. выполнение операции фильтрации занимает CEIL(ORDER/2) тактов на каждый отсчет выходного сигнала.

Блок имеет два конфигурируемых профиля. Профиль – набор коэффициентов и параметров фильтра (порядок фильтра, коэффициент интерполяции, масштабирования). Для каждого

из профилей предусмотрен отдельный набор регистров управления. В каждый момент времени задействован один из профилей. Профили могут перепрограммироваться и переключаться в ходе работы. Таким образом, можно запрограммировать неактивный профиль во время работы активного и переключиться между ними прямой командой или по сигналу SYNC_SFG. Переключение профилей задерживается автоматически до окончания вычисления очередного отсчета фильтра, что исключает порчу данных при переключении.

Поле PROFILE регистра IFIRx_CFG (x – номер устройства IFIR64, здесь и далее в разделе) отражает при чтении активный в настоящий момент профиль. Запись в поле приводит формированию запроса на переключение. Поле SYNCCFGSENCE включает (1) синхронизацию переключения профилей с SYNC_CFG.

Коэффициенты фильтра записываются в регистры с наименованием IFIRx_Py_CFk, где y заменяется на номер профиля, k – на номер коэффициента. Остальные параметры профиля управляются полями регистра с наименованием IFIRx_PROFILEy.

При записи коэффициентов в регистры 24-битное значение в 32-битном слове должно быть выровнено по младшему биту. При чтении коэффициентов из регистров они расширяются знаком до 32 бит.

В режимах симметричной и антисимметричной характеристики фильтров в памяти содержится только половина коэффициентов. При использовании нечетного порядка фильтра центральный коэффициент должен быть записан в регистр как ½ своего значения. При порядках фильтра менее предельных коэффициенты загружаются в регистры с младшими номерами, а незадействованные регистры заполнять не обязательно.

Память коэффициентов не определена при включении питания и не сбрасывается по сигналу сброса, поэтому подлежит обязательной инициализации перед включением фильтра в работу.

Поле SYM выбирает вид симметрии характеристики (симметричная при SYM=0, антисимметричная при SYM=1, несимметричная при SYM=2,3).

Поле ORDER задает порядок фильтра. Число отводов фильтра равно ORDER+1.

Поле DR задает коэффициент децимации, который устанавливается равным DR+1.

Поле IR определяет количество добавляемых отсчетов при интерполяции.

Поле SCALE задает коэффициент масштабирования, определяя величину сдвига влево значений 47-разрядных аккумуляторов фильтра перед взятием из них старших 18 бит для подачи на выход.

21.3.1.9.4.3 Полуполосные фильтры-интерполяторы IHB

Блок полуполосных фильтров состоит из трех включенных последовательно КИХ фильтров-интерполяторов с постоянными коэффициентами IHB3, IHB2 и IHB1 (именно в таком порядке), каждый из которых может осуществлять децимацию в 2 раза либо работать в прозрачном режиме (bypass).

Все перечисленные устройства управляются через регистр IHB_MODE. При записи единицы в поле CLR регистра управления происходит сброс счетчика децимации и очистка данных в тракте вычислений.

Поле IHB_MODE включает и выключает каскады. Фильтры включаются последовательно, начиная с DHB3. Т.е. нельзя включить DHB2, не включив DHB3, и нельзя включить DHB1, не включив DHB2 и DHB3.

Устройства масштабирования на выходах фильтров не предусмотрены. Коэффициент усиления фильтров по постоянной составляющей близок к 1.

К выходу каждого устройства масштабирования подключен блок оценки уровня сигнала BWGAUGE. Полученные оценки уровня сигнала отображаются в регистре IHB_BWGAUGE. Операция записи в регистр IHB_BWGAUGE приводит к сбросу оценок и блоков BWGAUGE и началу нового цикла оценки.

Коэффициенты фильтров приведены в таблице.

Таблица 21.175. Коэффициенты IHB фильтров

IHB1	IHB2	IHB3
		2
		0
		-14
		0
		62
		0
		-204
		0
	4	547
	0	0
	-35	-1271
	0	0
13	166	2721
0	0	0
-103	-583	-5972
0	0	0
602	2496	20513
1024	4096	32768
602	2496	20513
0	0	0
-103	-583	-5972
0	0	0
13	166	2721
	0	0
	-35	-1271
	0	0

IHB1	IHB2	IHB3
	4	547
		0
		-204
		0
		62
		0
		-14
		0
		2

АЧХ фильтров представлены на рисунках ниже.

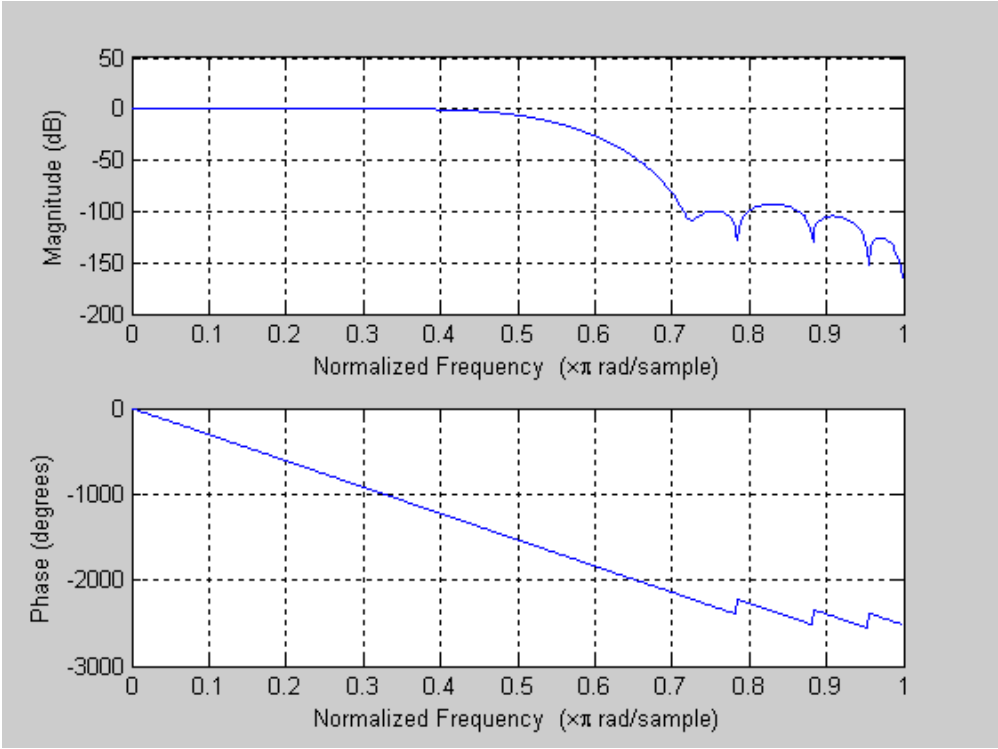


Рисунок 21.47. АЧХ IHB3 фильтра

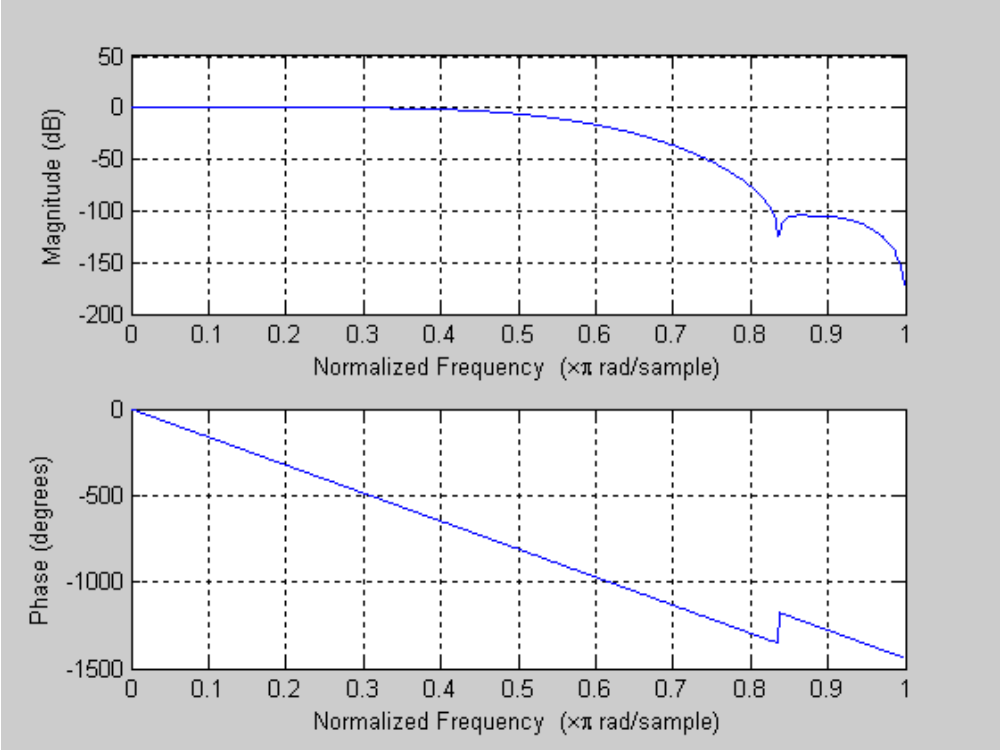


Рисунок 21.48. АЧХ IHB2 фильтра

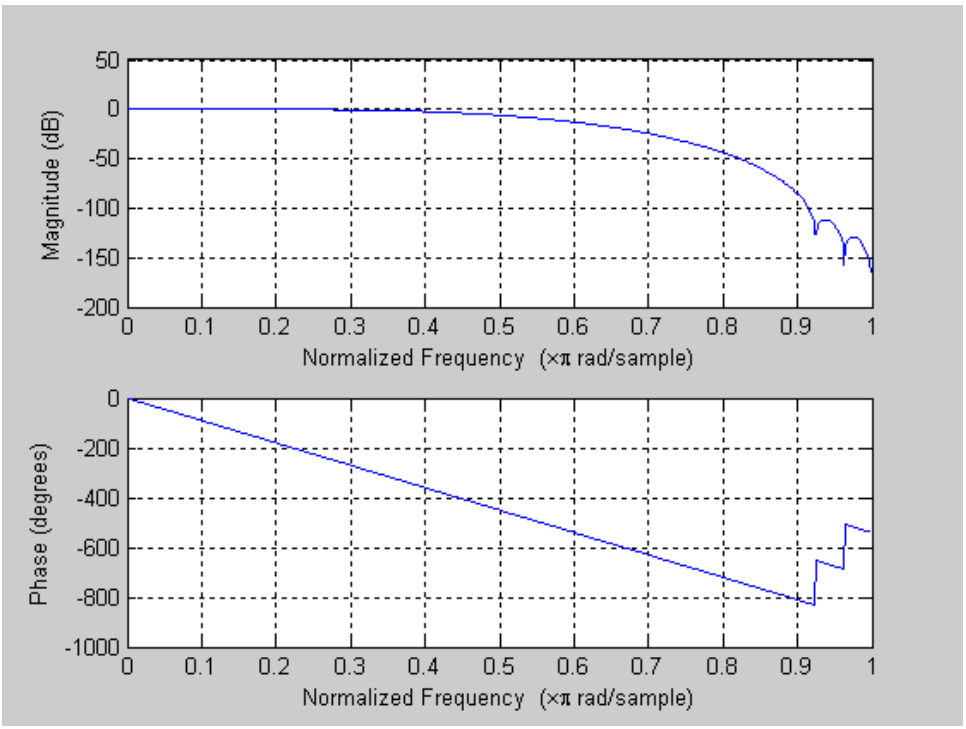


Рисунок 21.49. АЧХ IHB1 фильтра

21.3.1.9.4.4 ICIC интерполяторы

Блок включает в себя два каскада интерполяции и фильтрации – ICICN и ICIC2. Максимальный коэффициент интерполяции в каскаде CIC2 – 128 раз. Максимальный

коэффициент интерполяции в каскаде CICN зависит от конфигурации и ограничен величинами:

- для CIC3 в 128 раз
- для CIC4 в 100 раза,
- для CIC5 в 32 раз,
- для CIC6 в 16 раз.

Обобщенная блок-схема ICIC интерполятора приведена на Рисунок 21.50.

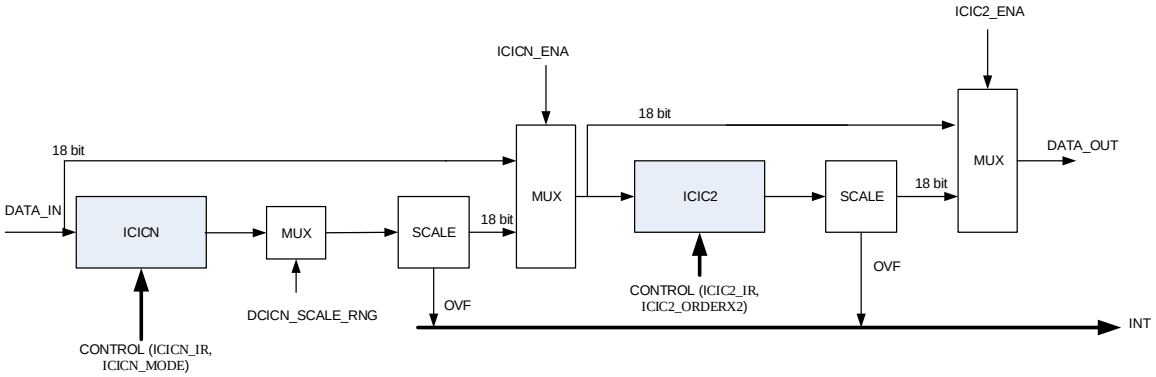


Рисунок 21.50. Блок-схема ICIC интерполятора

После каждого из фильтров включено устройство масштабирования, приводящее разрядность сигнала к общему ограничению тракта обработки в 18 бит.

Фильтры и устройства масштабирования управляются полями регистров ICICN_CFG и ICIC2_CFG аналогично фильтрам-дециматорам приемного тракта. Отличаются лишь разрядности сигналов на выходах фильтров и общем выходе устройства. См. описание регистров ICICN_CFG и ICIC2_CFG и описание работы DCIC дециматоров.

К выходу каждого мультиплексора подключен блок оценки уровня сигнала BWGAUGE. BWGAUGE анализирует заполненность разрядной сетки – количество задействованых бит на интервале анализа. Полученные оценки уровня сигнала отображаются в регистре ICIC_BWGAUGE. Операция записи в регистр ICIC_BWGAUGE приводит к сбросу оценок и началу нового цикла оценки. Оценка – положительное целое число, указывающее на первый значащий бит.

21.3.1.9.4.5 Гетеродин TX_HTRD

Блок осуществляет перенос частоты комплексного сигнала на промежуточную или несущую частоту. Возможности и управление блоком практически полностью идентичны таковым для гетеродина приемной части. Разница заключается в разрядности сигнала, которая составляет 2x18 бит для гетеродина передающего канала.

На выходе гетеродина установлен фильтр INVSINC – инверсный sin(x)/x, предназначенный для сглаживания частотной характеристики выхода АЦП. Фильтр включается установкой в «1» бита INVSINC_ENA в общем регистре управления канала TXC_CHNL_CTRL.

21.3.2 DFE_NELS

21.3.2.1 Система управления

Управление блоком DFE осуществляется через интерфейс APB и через интерфейсы AXI (Command FIFO и RX/TX FIFO). Интерфейс APB служит для общего управления блоком, интерфейсы Command FIFO служат для синхронного выполнения команд через DFE commander.

Через интерфейс APB за одну команду передаются и принимаются 32-битный адрес и данные. Через интерфейс доступны следующие функции:

- Отправка команд в модули DFE_Commander с немедленным выполнением
- Управление регистрами не требующими синхронного управления
- Чтение данных DFE_Commander
- Считывание данных эстиматоров

Через интерфейсы Command FIFO доступна только загрузка команд DFE_Commander.

Дополнительно при использовании FIFO доступна команда WAIT и благодаря этому возможно синхронное выполнение команд при заданном значении таймера.

Список и описание регистров представлены в Разделе «Регистры блока DFE_NELS»

21.3.2.1.1 DFE Commander

Каждый канал DFE Commander предназначен для управления одним трактом передачи и группой из 4-х приемных трактов, а также одним из 4-х выделенных интерфейсов SPI и общими 16-ю выводами GPIO, подключенными к блоку DFE.

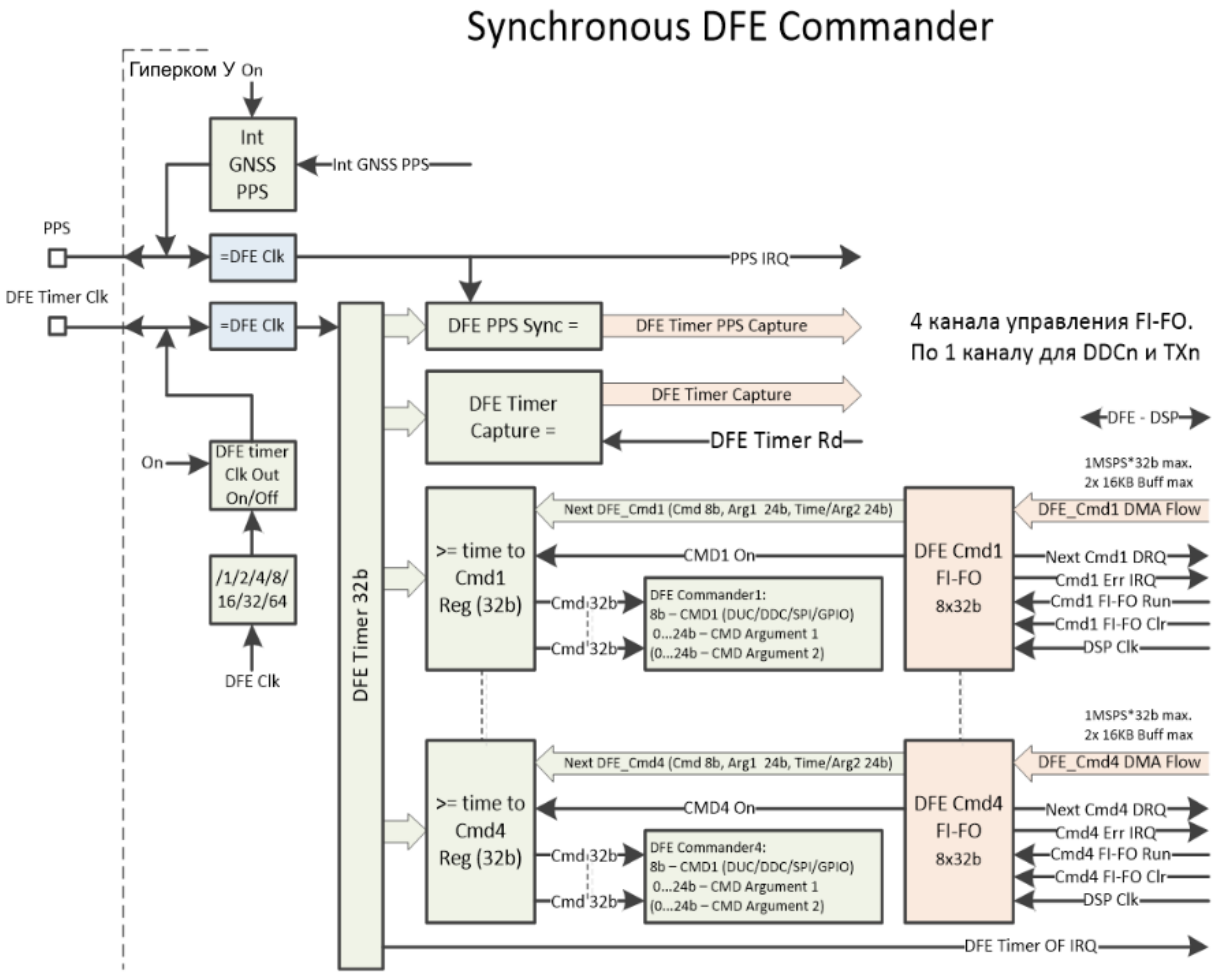


Рисунок 21.51. Synchronous DFE Commander

Модуль Synchronous DFE Commander обеспечивает синхронное управление с точной привязкой ко времени следующими процессами и модулями:

- старт/стоп(очистка) блоков DDC, DUC и др.;
- установка параметров частоты, скорости изменения частоты и начальной фазы в модули NCO Up/Down Mixers;
- синхронное конфигурирование потоков данных через мультиплексоры и сумматоры;
- синхронная смена буферов параметров и коэффициентов в модулях CFR, FIR и DPD;
- точное измерение времени фронта внутреннего или внешнего сигнала секундной метки PPS;
- получение временной привязки из блока DFE в DSP для определения точного времени прихода или выдачи блоков данных через каналы DMA;
- синхронное программирование внешних ИМС синтезаторов частоты, модуляторов/демодуляторов и других ИМС радиочастотного тракта посредством выделенных интерфейсов SPI блока DFE;

- синхронное управление сигналами GPIO блока DFE.

Так же в модуле Synchronous DFE Commander формируются сигналы внутренней тактовой частоты для синхронного приема/выдачи данных от/в АЦП/ЦАП.

DFE Commander по каждой из 4-х очередей получает управляющие 32-битные слова от ЦП по интерфейсу AMBA 3 APB или AXI.

Команды по AMBA 3 APB выполняются сразу после получения, команды по AXI помещаются в очередь команд для последующего синхронного выполнения.

Формат слова:

31	30.. 24	23:0
RESERVED	CMD CODE	DATA

CMD CODE – код команды.

Таблица 21.176.

Команда	Код	BUF	Данные	Доп. данные	Описание
x		x	X	x	Все незанятые коды взводят флаг ошибки WRONG_CMD
EXTRA_DATA	01	0,1	(23:0) доп. данные	нет	Доп. данные для след. команды
CLR_CMD_FIFO	02	0	Нет	нет	Очистка очереди команд
WAIT	03	1	значение таймера T(23:0)	значение таймера T(31:24)	Ожидание значения таймера t>=T
WRITE_DDUC	04	0,1	данные(23:0)	адрес(15:0), данные(31:24)	Запись в регистр канала DDC или DUC (см. таблицу адресов)
RX_CLR_RUN	05	0,1	(3:0) – маска: ‘0’ – сброс канала, ‘1’ – работа канала	нет	Синхронное управление 4-мя каналами DDC
RX_DDS_LOAD	07	0,1	(3:0) – маска загрузки DDS из временных регистров	нет	Синхронное загрузка 4-х каналов DDC

Команда	Код	BUF	Данные	Доп. данные	Описание
TX_CLR_RUN	08	0,1	(0) – ‘0’ –сброс DUC, ‘1’ – работа DUC (3:2) – CFR clr/run/hld/bp (5:4) – Usamp CLR/RUN/HLD/BP (7:6) – DPD CLR/RUN/HLD/BP (9:8) – Umix CLR/RUN/HLD/BP (11:10) – DAC corr CLR/RUN/HLD/BP	нет	Управление ПРД
WRITE_SPI	09	0,1	данные (23:0)	7’0, длина(1:0), скорость(2:0), CS(3:0), данные (31:24)	Вывод в SPI. Длина: 00 – 8 бит, 01 – 16 бит, 10 – 24 бит, 11 – 32 бит. Скорость: 000 – DFEclk/8 001 – DFEclk/16 … 111 – DFEclk/1024
ACCESS_GPIO	0A	0,1	маска (23:0)	8’0, тэг(3:0), тип доступа (3:0), маска (31:24)	Запись и чтение GPIO. Тип доступа: 0000 – AND_DATA 0001 – OR_DATA 0010 – AND_DIR 0011 – OR_DIR 0100 - READ

Для загрузки регистра через DFE commander используются команды 01(дополнительные данные) и 04(загрузка регистра).

Пример загрузки данных 0x12345678 в регистр по адресу 0xFEDC в канал 1:

	apb_addr	apb_data
1	00000004	01fedc12
2	00000004	04345678

21.3.2.1.2 Таймер

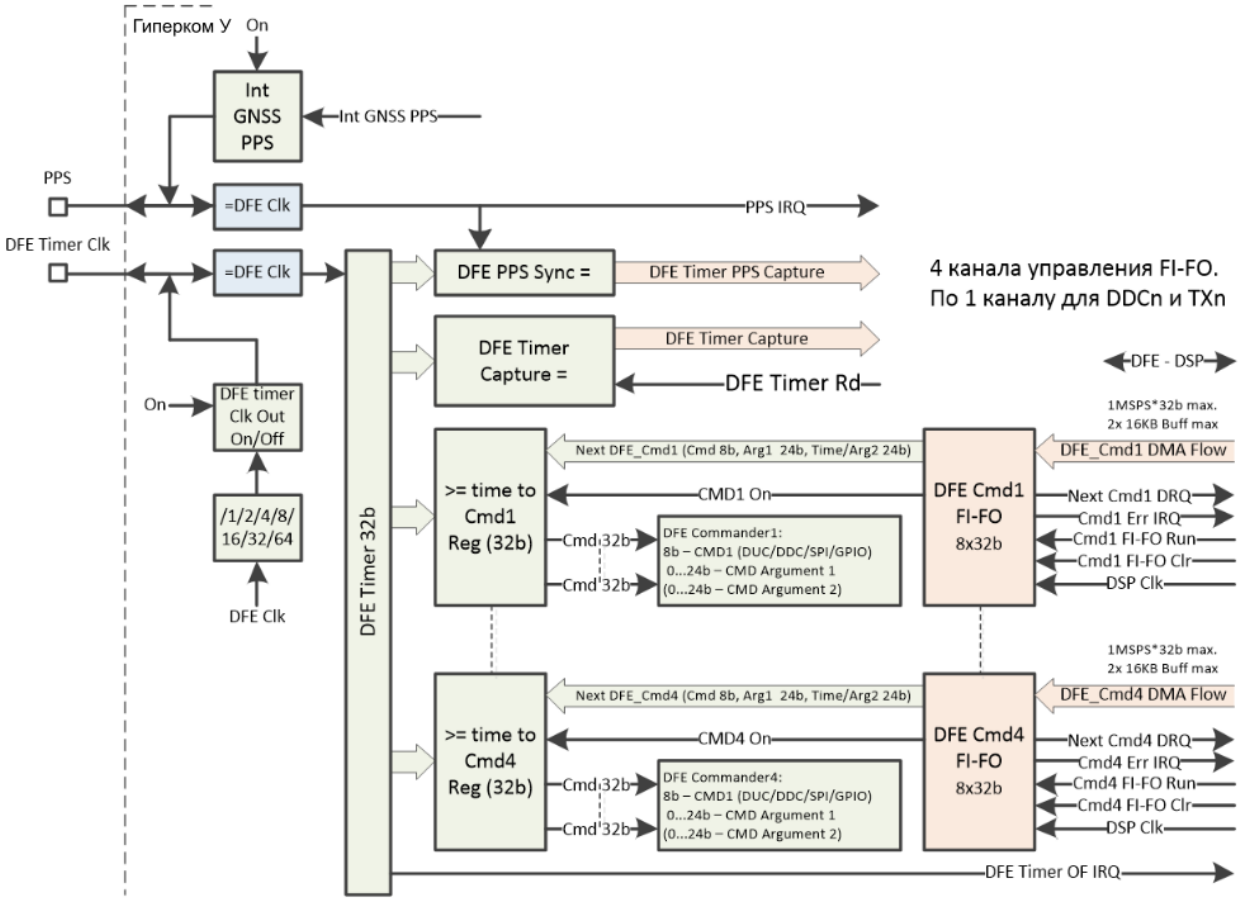


Рисунок 21.52. Таймер

Блок «таймер» служит внутренними часами блока dfe: он обеспечивает возможность синхронного исполнения команд в dfe commander и захват определенного момента времени.

Управляется только через DFE commander 4 (регистр “Timer”), доступна настройка частоты переключения счетчика и переключение направления порта DFE Timer clk

21.3.2.1.3 Интерфейсы SPI и GPIO

Для управления ИМС внешних радиотрактов (до 4-х радиотрактов) системы блок DFE использует до 4-х собственных раздельных интерфейсов SPI, каждый их которых работает синхронно по тактовому сигналу DFE Clk и содержит следующий набор из 7-ми сигналов:

- MOSI_n – сигнал данных, передаваемых от мастера к слэйву. Индекс “n” от 1 до 4-х определяет номер канала SPI;
- MISO_n – сигнал данных, принимаемых мастером от слэйва;
- SPI_Clk_n – сигнал синхронизации передачи данных. Данные выдаются источником по отрицательному фронту сигнала SPI_Clk_n, а принимаются (фиксируются) приемником по положительному фронту. Значение частоты сигнала SPI_Clk_n, в

пределах примерно от 100 кГц до 100 МГц, определяется программируемым синхронным делителем частоты DFE Clk индивидуально для каждого канала SPI;

- SPI_CS_n.m – (где “m” = 1, 2, 4, 8) сигнал «Chip Select», нулевой уровень на котором определяет один из 4-х подканалов к «слэйвам». Количество подканалов может быть увеличено до 15-ти, если к выходам SPI_CS_n.m подключить дешифратор 1:2:4:8 в 16. При этом первые 15 выходных сигналов дешифратора будут определять номер активного подканала SPI интерфейса.

Любые сигналы каждого из 4-х SPI интерфейсов могут быть запрограммированы в режим сигналов общего назначения GPIO_n, режим работы и значения которых могут быть заданы или считаны посредством команд DSP через Synchronous DFE Commander.

21.3.2.1.4 Тактирование каналов приёмника и передатчика

Для контроля энергопотребления схемы возможно отключение и включение тактовых сигналов раздельно на каждом канале приёмника и передатчика.

21.3.2.1.5 Эстиматоры

Эстиматоры предназначены для оценки параметров сигнала. В схеме присутствуют следующие эстиматоры:

- Эстиматор на каждый канал приёмника. Предназначены для оценки данных загружаемых в выходное fifo данных. Оцениваются сумма квадратов загружаемых данных $(I^2 + Q^2)_{max}$ и сумма квадратов за промежуток времени $\sum(I^2 + Q^2)$

- Эстиматор канала bypass. Параметры считаются для сигнала с АЦП, оценивается сумма квадратов загружаемых данных A^2_{max} и сумма квадратов за промежуток времени $\sum A^2$

- Эстиматор на каждый канал передатчика. Сигнал снимается после модуля UP Mixer. Оцениваются параметры $(I^2 + Q^2)_{max}$, I^2_{max}, Q^2_{max}, $\sum(I^2 + Q^2)$

21.3.2.2 Приёмник

21.3.2.2.1 Demapper

Модуль осуществляет переход между клоковыми доменами с использованием FIFO, что позволяет работать с частотами DFE равными или большими, чем частота сэмплирования АЦП.

При работе в режиме FIFO данные из АЦП записываются по поднимающимся и спадающим фронтам клока в соответствующие регистры FIFO. Перед запуском системы производится сброс FIFO, гарантирующий детерминированность начального состояния FIFO, а также начало записи по поднимающемуся, либо падающему фронту клока (в зависимости от используемого АЦП). В том случае, если ведется работа с шириной шины более 8 бит, на обе пары FIFO поступает общий сигнал частоты дискретизации, подаваемый от тактового входа первого интерфейса LVDS ADC1_CLK.

Домен DFE_CLK производит чтение FIFO при условии наличия данных, записанных в FIFO по переднему и по заднему фронтам.

При работе в режиме SDR вычитывание данных происходит только из одного из регистров FIFO, в зависимости от активного фронта.

Мультиплексоры осуществляют выбор входных данных в соответствии с полями управляющих регистров, определяющих активную фазу сигнала дискретизации, режим работы DDR/SDR, порядок следования битов и т.д.

На Рисунок 21.53 показан модуль интерфейсов LVDS и JESD204B для подключения ИМС АЦП блоку DFE. Более подробная структурная схема модуля интерфейсов АЦП приведена на Рисунок 21.54.

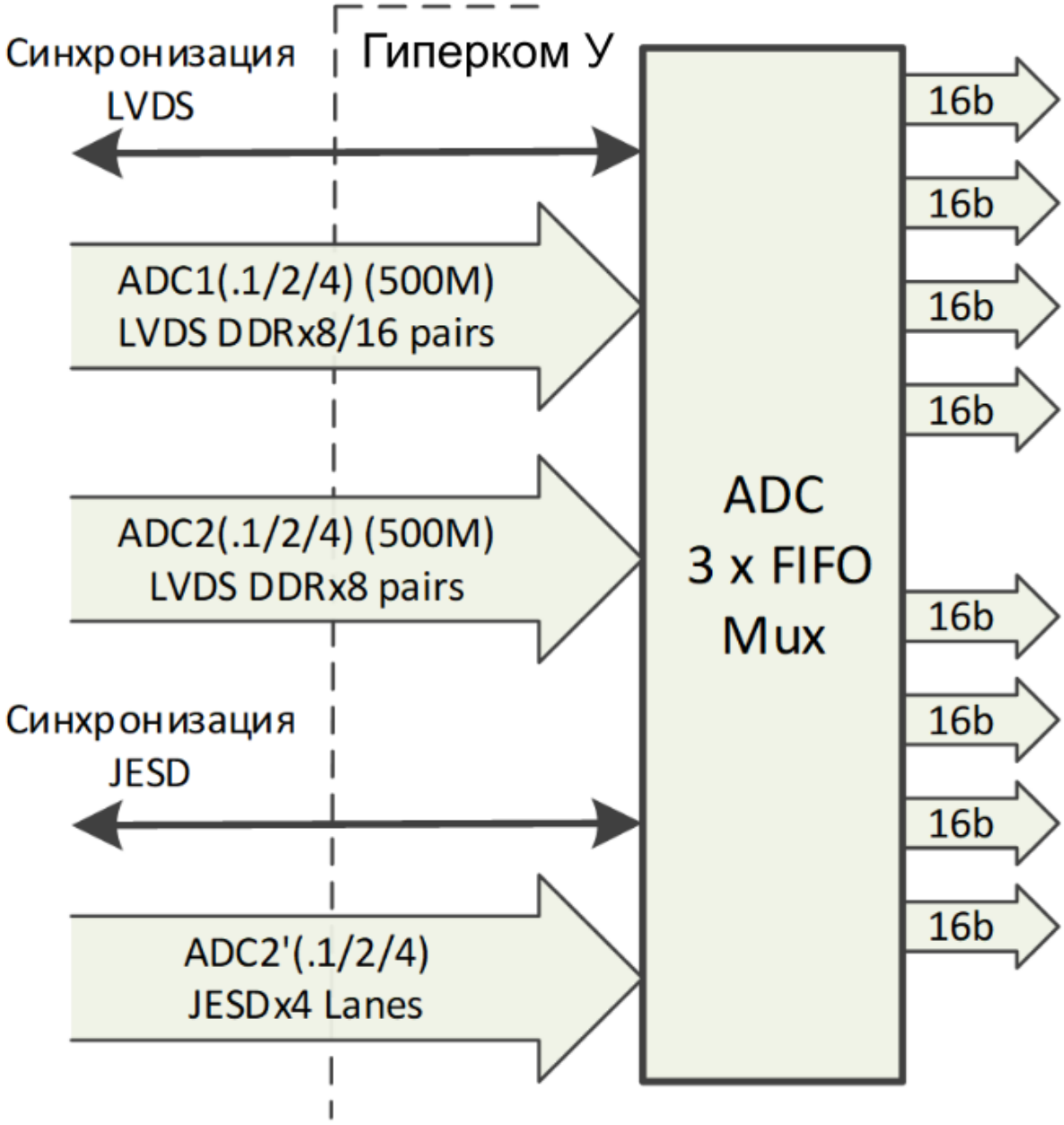


Рисунок 21.53. Модуль интерфейсов LVDS и JESD204B для подключения ИМС АЦП блоку DFE

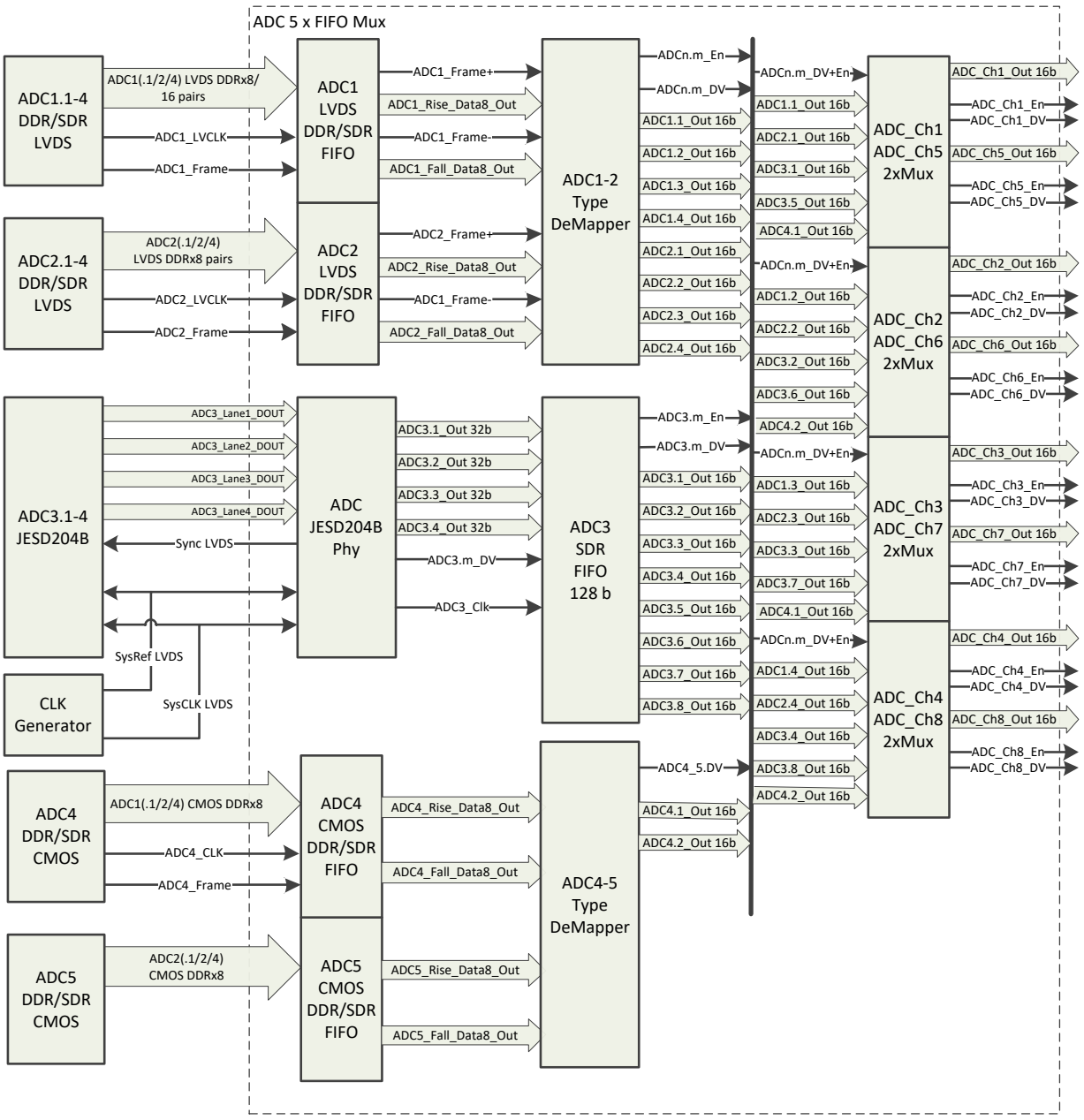


Рисунок 21.54. Структурная схема модуля интерфейсов подключения АЦП с FIFO и выходными мультиплексорами

При работе с АЦП, подключаемыми по 8-ми битной шине LVDS, схема обеспечивает независимую работу двух таких АЦП, каждый их которых может иметь собственный источник тактового сигнала. При подключении АЦП с шиной данных от 10 до 16-ти, схема обеспечивает синхронную работу обоих модулей LVDS DDR/SDR ADC FIFO.

Режим работы схемы определяется программированием сигнала ADC1_8/16 в регистре управления модуля. Единичное значение этого бита определяет режим независимой работы двух модулей LVDS DDR/SDR ADC FIFO с разными АЦП. Модуль может функционировать как в режиме SDR, так и DDR.

Управление блоком осуществляется через регистры “ADC_Demapper_conf” и “ADC_Demapper_mode”.

21.3.2.2.1.1 Примеры подключения АЦП

21.3.2.2.1.1.1 ИМС АЦП LTC2262-14

- тип интерфейса - LVDS DDR 7 бит;
- биты АЦП 0,2,4,6,8,10,12, передаваемые по rising edge, считываются из старших битов регистра ADC1_RISE_DATA_OUT;
- биты АЦП 1,3,5,7,9,11,13, передаваемые по falling edge, считываются из старших битов регистра ADC1_FALL_DATA_OUT;
- вывод d0 интерфейса ADC1 не подключается;
- на выходе DeMapper биту Dout15 присваивается значение Din13. Биты Dout 1:0 =0;
- можно подключить второй АЦП LTC2262-14/12 на вход ADC2 и считывать данные из регистров ADC2_RISE_DATA_OUT и ADC2_FALL_DATA_OUT аналогично первому АЦП.
- АЦП необходимо подключать к портам adc1_din(7:1) и adc2_din(7:1)

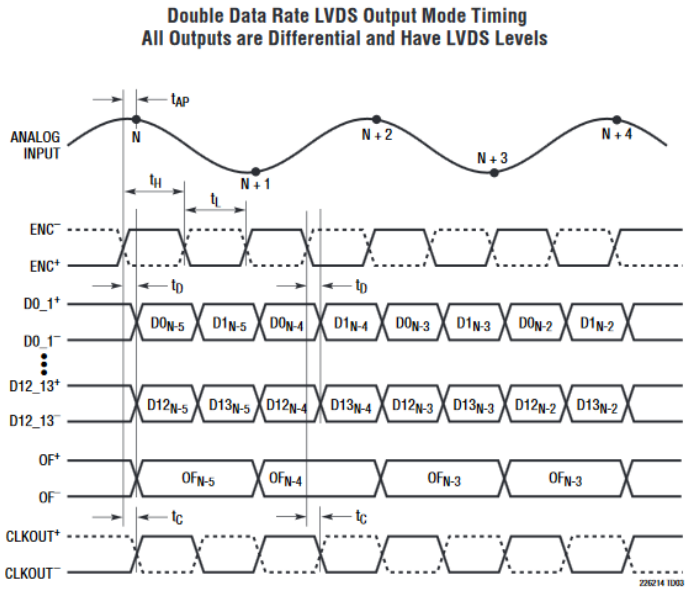


Рисунок 21.55

В регистры управления пишем следующие значения:

ADC_Demapper_conf == 32’h0000_2000

ADC_Demapper_mode == 32’h1F00_1000

21.3.2.2.1.1.2 ИМС АЦП LTC2262-12

- тип интерфейса - LVDS DDR 6 бит;

- бит управления типом разрядности ADC_8/16=1;
- биты АЦП 0,2,4,6,8,10, передаваемые по rising edge, считываются из старших битов регистра ADC1_RISE_DATA_OUT;
- биты АЦП 1,3,5,7,9,11, передаваемые по falling edge, считываются из старших битов регистра ADC1_FALL_DATA_OUT;
- выводы d0, d1 интерфейса ADC1 не подключается;
- на выходе DeMapper биту Dout15 присваивается значение Din11. Биты Dout 3:0 =0;
- АЦП необходимо подключать к портам adc1_din(7:2) и adc2_din(7:2)

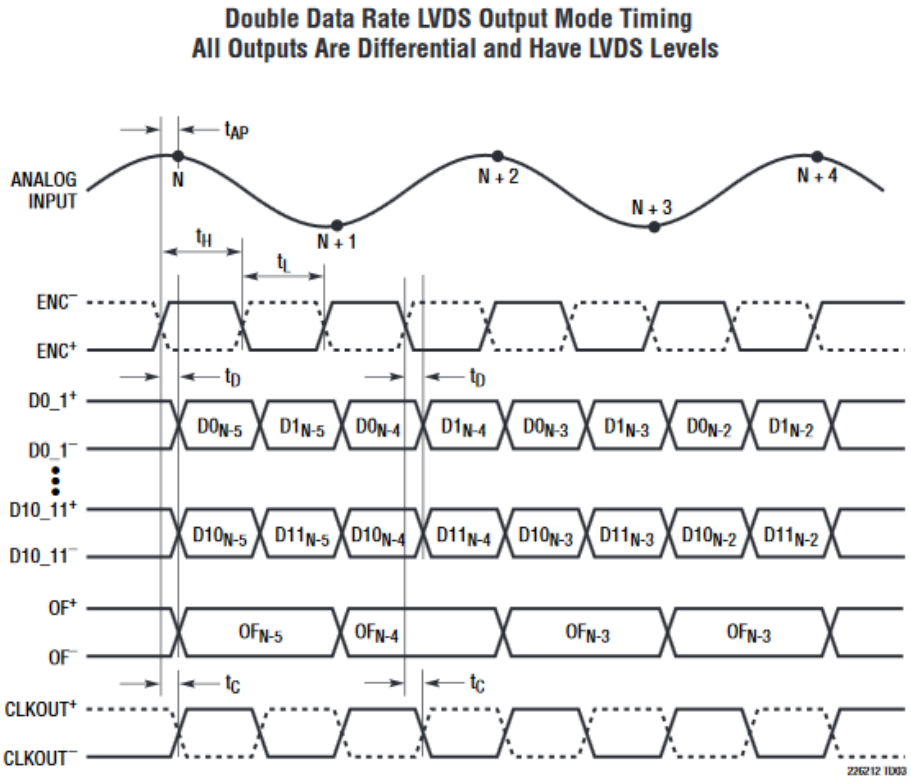


Рисунок 21.56

В регистры управления пишем следующие значения:

ADC_Demapper_conf == 32’h0000_0000

ADC_Demapper_mode == 32’h1F00_1000

21.3.2.2.1.1.3 ИМС АЦП AD9434/AD9684 (режим 1)

- LVDS 12/14,(16) бит;
- биты Din 7:0 регистра ADC1_RISE_DATA_OUT содержат младшие 6/7(8) бит АЦП;
- биты Din 15:8 регистра ADC2_RISE_DATA_OUT содержат старшие 6/7(8) бит АЦП;
- на выходе Dout15=Din11/13(15). Биты Dout 3/1:0 =0;

- старшие биты АЦП подключаются к старшим битам интерфеса ADC2;
- младшие неиспользуемые выводы ADC1 обнуляются на выходе, число обнуляемых младших бит задается - 0/2/4.

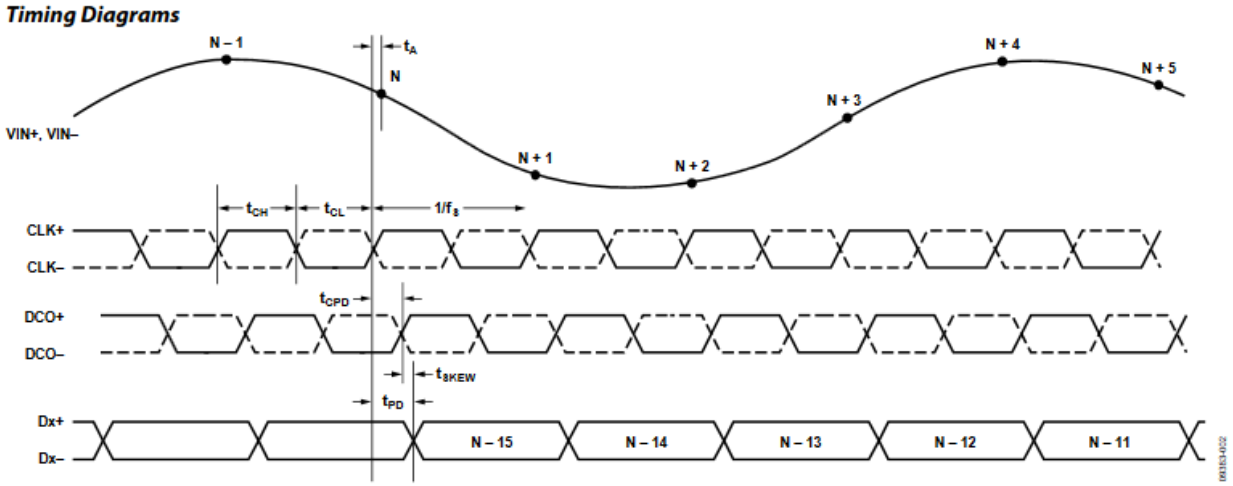


Рисунок 21.57

В регистры управления пишем следующие значения:

ADC_Demapper_conf == 32’h0000_2000

ADC_Demapper_mode == 32’h1F00_1000

21.3.2.2.1.1.4 ИМС АЦП AD9434

- LVDS DDR 12 бит;
- биты ADC1 Din 2:7 регистра ADC1_RISE_DATA_OUT содержат младшие 6 бит АЦП;
- биты ADC1 Din 10:15 регистра ADC1_FALL_DATA_OUT содержат старшие 6 бит АЦП;
- младшие 2 бита интерфейсов ADC1 и ADC2 не подключаются.

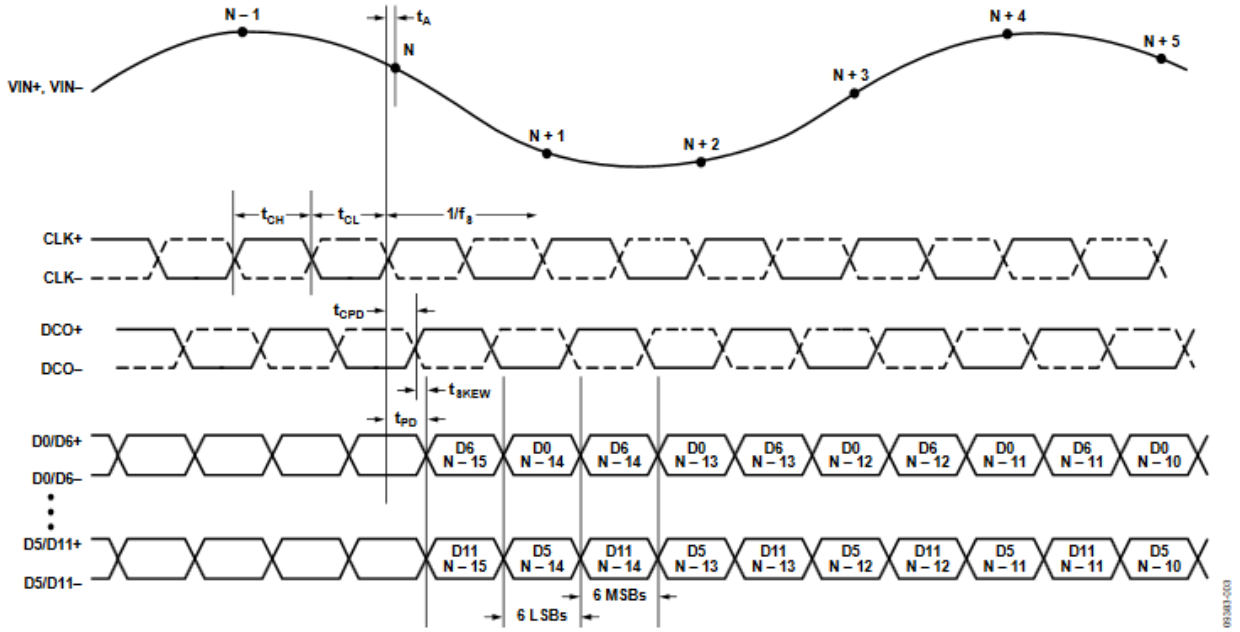


Figure 3. Double Data Rate Mode

Рисунок 21.58

В регистры управления пишем следующие значения:

ADC_Demapper_conf == 32’h0000_2040

ADC_Demapper_mode == 32’h1F00_1000

21.3.2.2.1.1.5 ИМС АЦП AD9650

- LVDS DDR 2x16 бит;
- биты ADC1 Din 0:7 регистра ADC1_RISE_DATA_OUT содержат младшие 8 бит АЦП1;
- биты ADC1 Din 0:7 регистра ADC1_FALL_DATA_OUT содержат младшие 8 бит АЦП2;
- биты ADC2 Din 0:7 регистра ADC2_RISE_DATA_OUT содержат старшие 8 бит АЦП1;
- биты ADC2 Din 0:7 регистра ADC2_FALL_DATA_OUT содержат старшие 8 бит АЦП2.

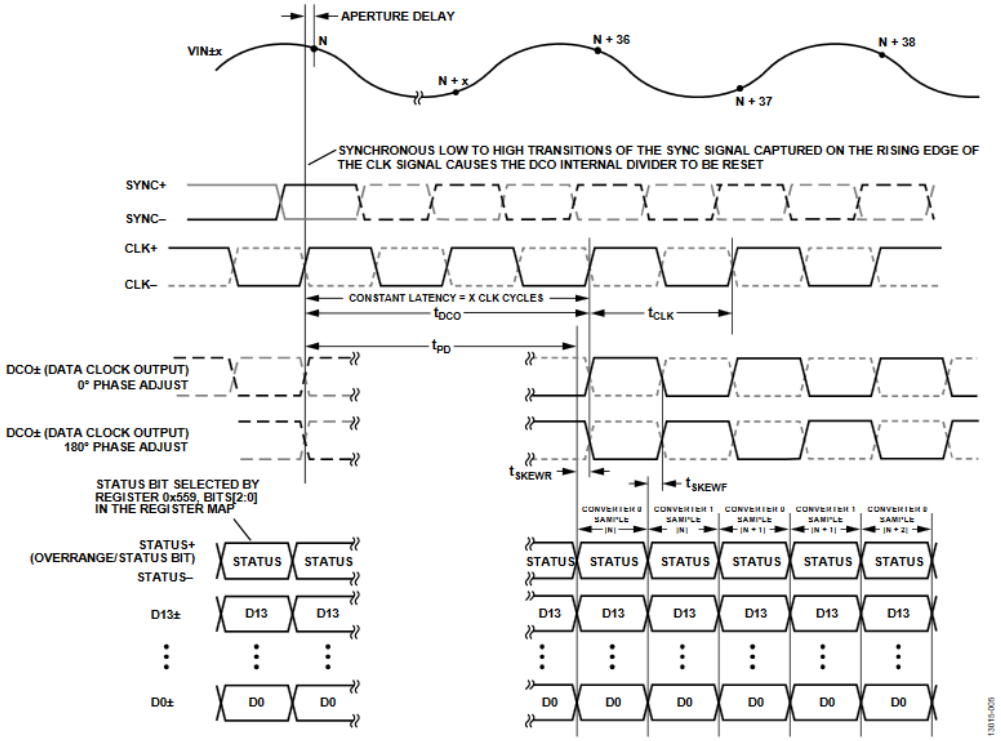


Figure 5. Parallel Interleaved Mode—Two Converters, ≤14-Bit Data, Output Sample Rate < 625 MSPS

Рисунок 21.59

В регистры управления пишем следующие значения:

ADC_Demapper_conf == 32’h0000_2100

ADC_Demapper_mode == 32’h1F00_1000

21.3.2.2.1.1.6 ИМС АЦП ADS4249/AD9684

- (режим 2), LVDS DDR 2x14 бит;
- подключается аналогично ADC LTC2262-14;
- первый канал подключается к старшим 7 бит ADC1;
- второй канал подключается к старшим 7 бит ADC2;
- выходные биты d1:0 обнуляются в обоих каналах АЦП;
- при инициализации необходимо запускать оба канала ADC1 и ADC2 одновременно по сигналам ADC1_RUN/CLR и ADC2_RUN/CLR.

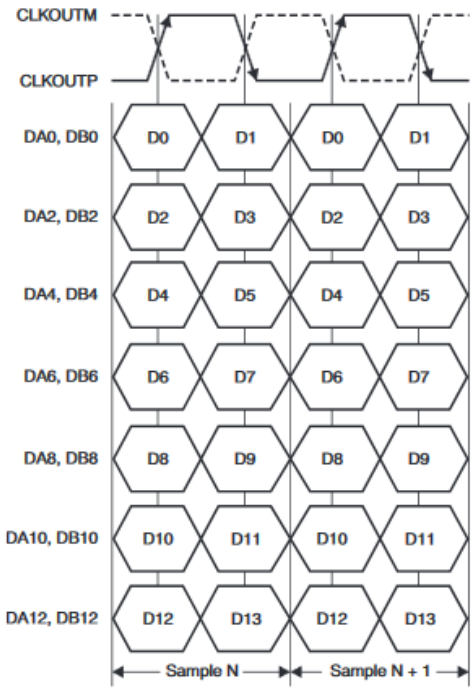


Figure 4. LVDS Interface Timing Diagram

Рисунок 21.60

В регистры управления пишем следующие значения:

ADC_Demapper_conf == 32’h0000_2140

ADC_Demapper_mode == 32’h1F24_9000

21.3.2.2.1.1.7 ИМС АЦП ADC3424

- LVDS DDR 4x12 бит;
- каждый канал передает 12 бит за 3 такта ADC_CLK по 2-м LVDS парам, четные биты по переднему фронту, нечетные по заднему;
- выходные 12 бит по каждому каналу прижимаются к старшим битам d15:d4, биты d3:d0 обнуляются.

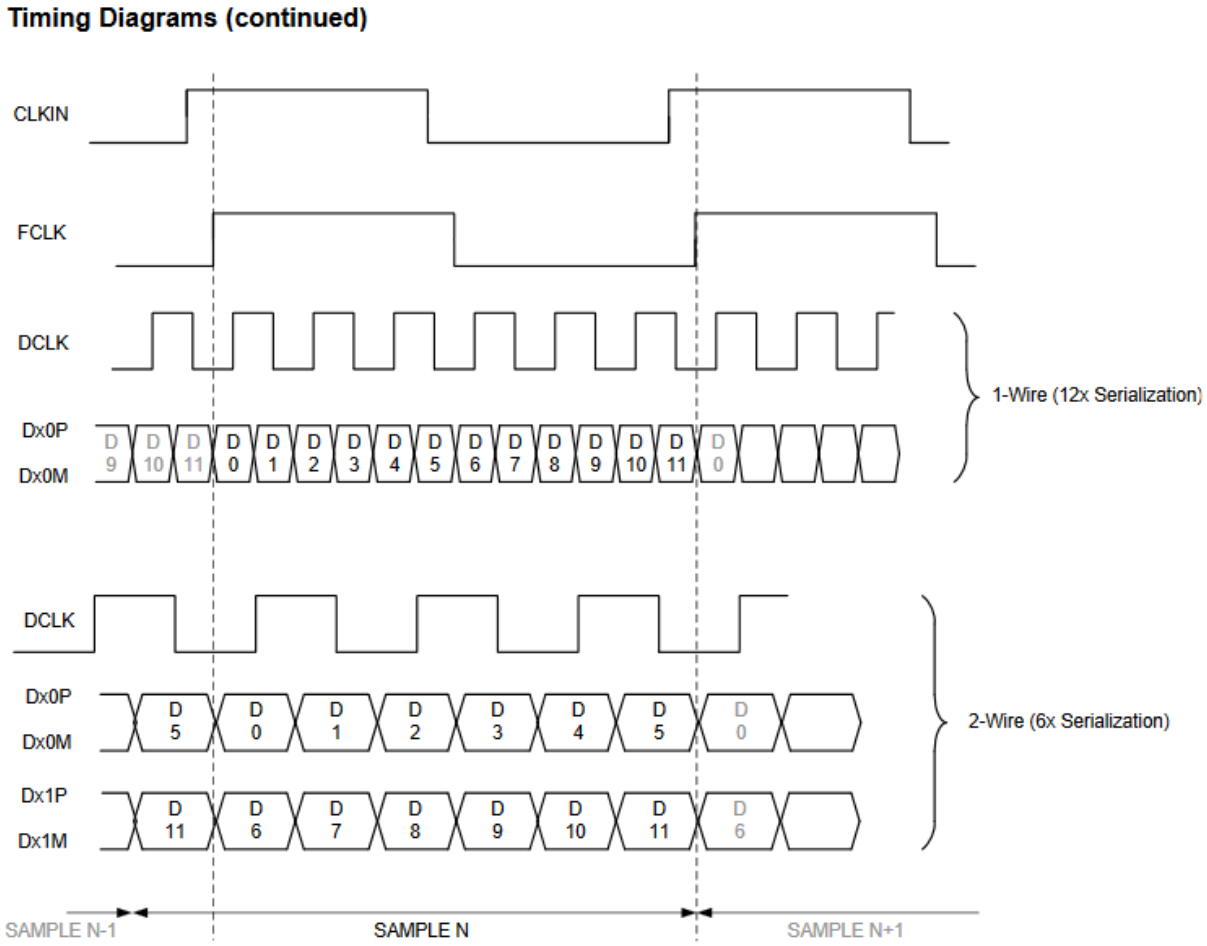


Рисунок 21.61

В регистры управления пишем следующие значения:

ADC_Demapper_conf == 32'h0000_1540

ADC_Demapper_mode == 32'h1F24_9000

21.3.2.2.1.1.8 ИМС АЦП ADC3544

- CMOS SDR/DDR 14 бит
- В SDR режиме один канал подключается к битам 0..13
- В DDR Режиме биты 13, 11, 9, 7, 5, 3, 1 берутся из RISE регистров и биты 12, 10, 8, 6, 4, 2, 0 берутся из FALL регистров

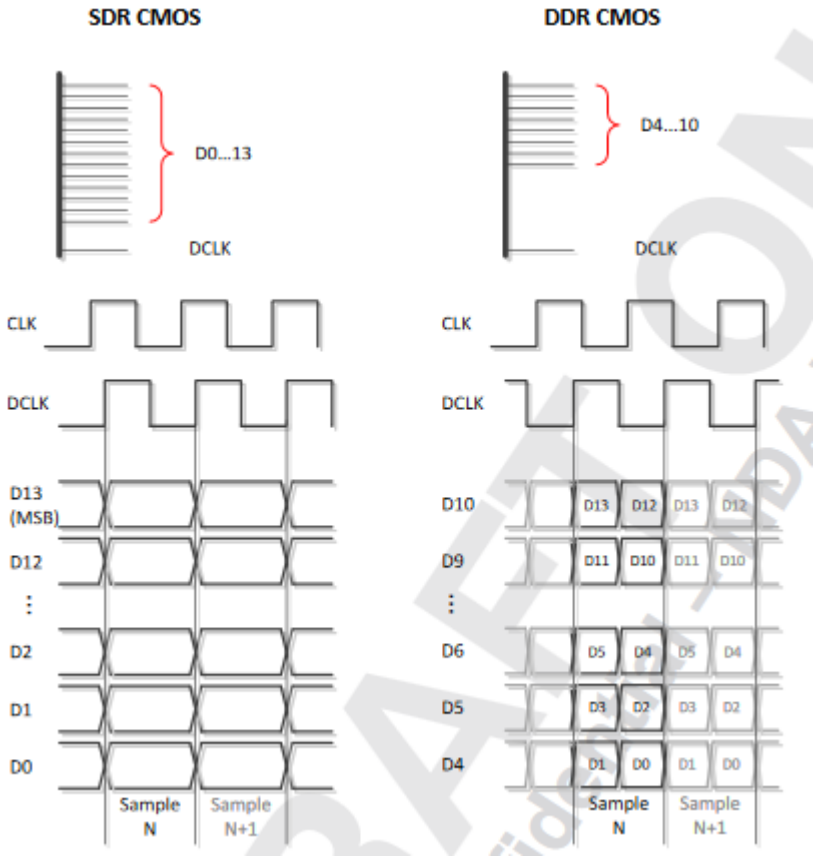


Рисунок 21.62

В регистры управления пишем следующие значения:

ADC_Demapper_conf == 32’h0000_0000

ADC_Demapper_mode == 32’h1F00_0004

21.3.2.2.1.1.9 ИМС АЦП ADC3644

- CMOS SDR/DDR
- В первом режиме для первого канала биты 13, 11, 9, 7, 5, 3, 1 передаются по ADC4_rise, биты 12, 10, 8, 6, 4, 2, 0 по ADC4_fall, для второго канала аналогично по ADC5
- Во втором режиме по rise передаются данные первого канала по rise, данные второго канала передаются по fall.

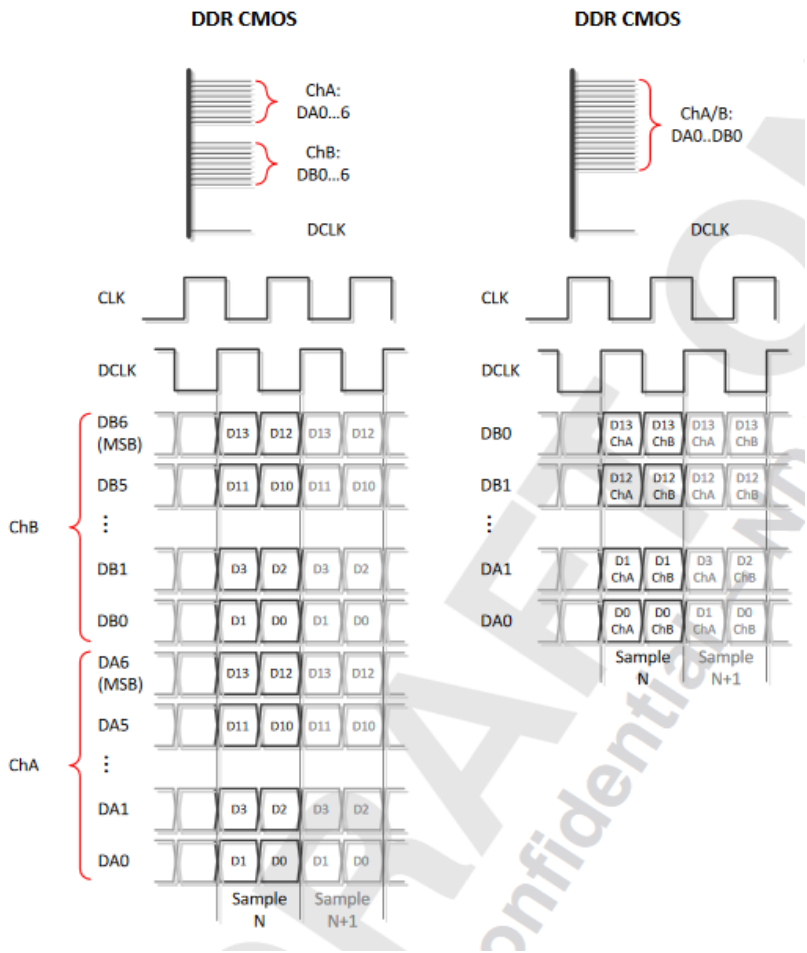


Рисунок 21.63

В регистры управления пишем следующие значения:

ADC_Demapper_conf == 32’h0001_0000

ADC_Demapper_mode == 32’h1F00_0024

21.3.2.2.2 Zero/Gain/Angle Correction

Вышедшие из ADC_FIFO 8 каналов данных объединяются попарно и обрабатываться в блоках Zero/Angle/Gain Correction в раздельном или IQ режиме.

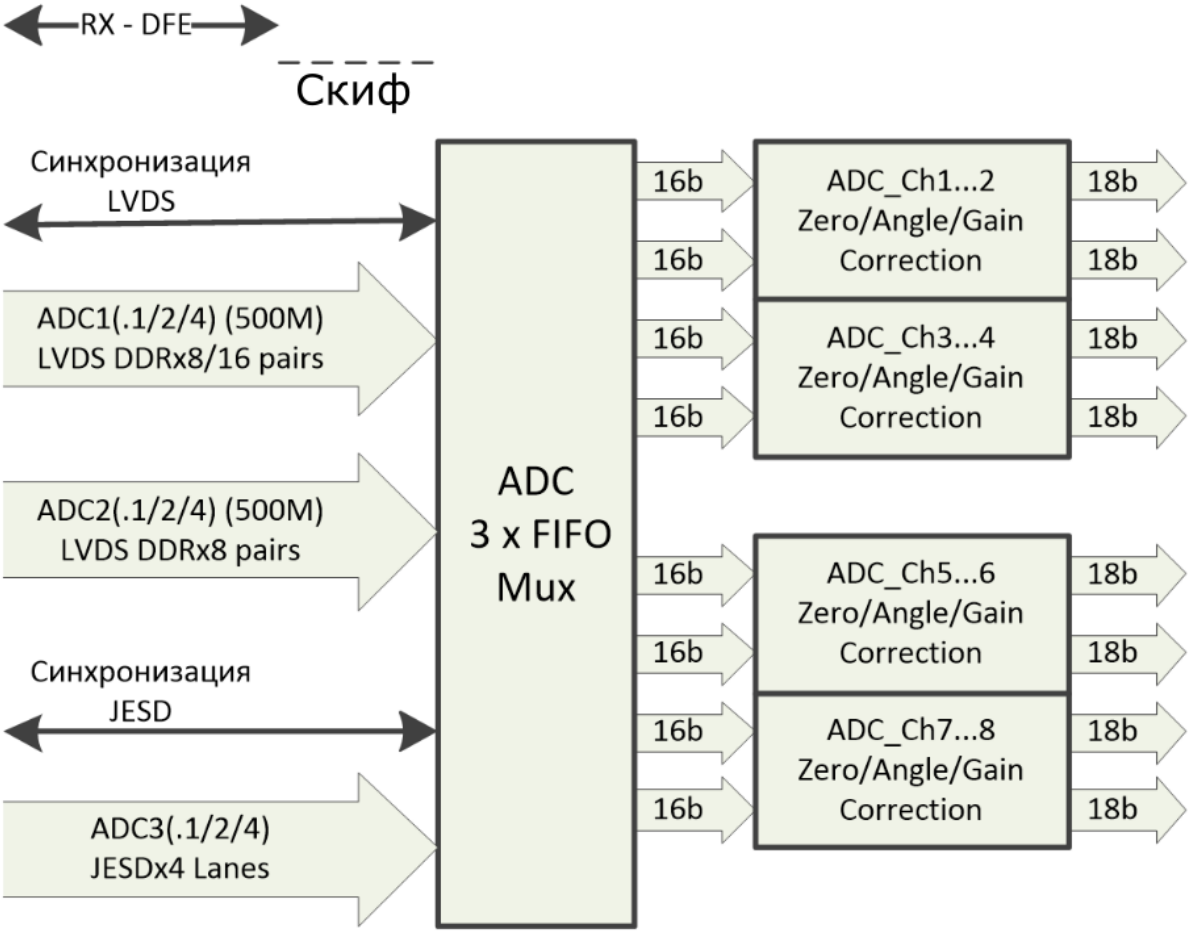


Рисунок 21.64

Каждый блок управляется собственным DFE commander через регистры «zero offset correction I RX», «zero offset correction Q RX», «Angle correction», «Gain correction I», «Gain correction Q», «Dither I», «Dither Q» и обеспечивает:

1. смещение нулей и добавление дитера раздельно по двум каналам
2. коррекцию неортогональности квадратурного сигнала
3. коррекцию разницы усиления по каналам

21.3.2.2.2.1 Коррекция смещения «нулей»

Модуль ADC[n]_Zero/Gain/Angle Correction обеспечивает коррекцию смещения «нулей», разницы усиления и неортогональности квадратурных составляющих комплексного сигнала от АЦП по каналам «I» и «Q».

Структурные схемы модулей коррекции нулей, коэффициентов усиления и неортогональности квадратурных сигналов показаны на Рисунок 21.65, Рисунок 21.66 и Рисунок 21.67 соответственно.

Синхронизация сэмплов данных от АЦП с тактовой частотой DFE_Clk блока DFE производится по сигналам синхронной выдачи данных DV(Data Valid) и разрешения синхронного приема данных ND (New Data).

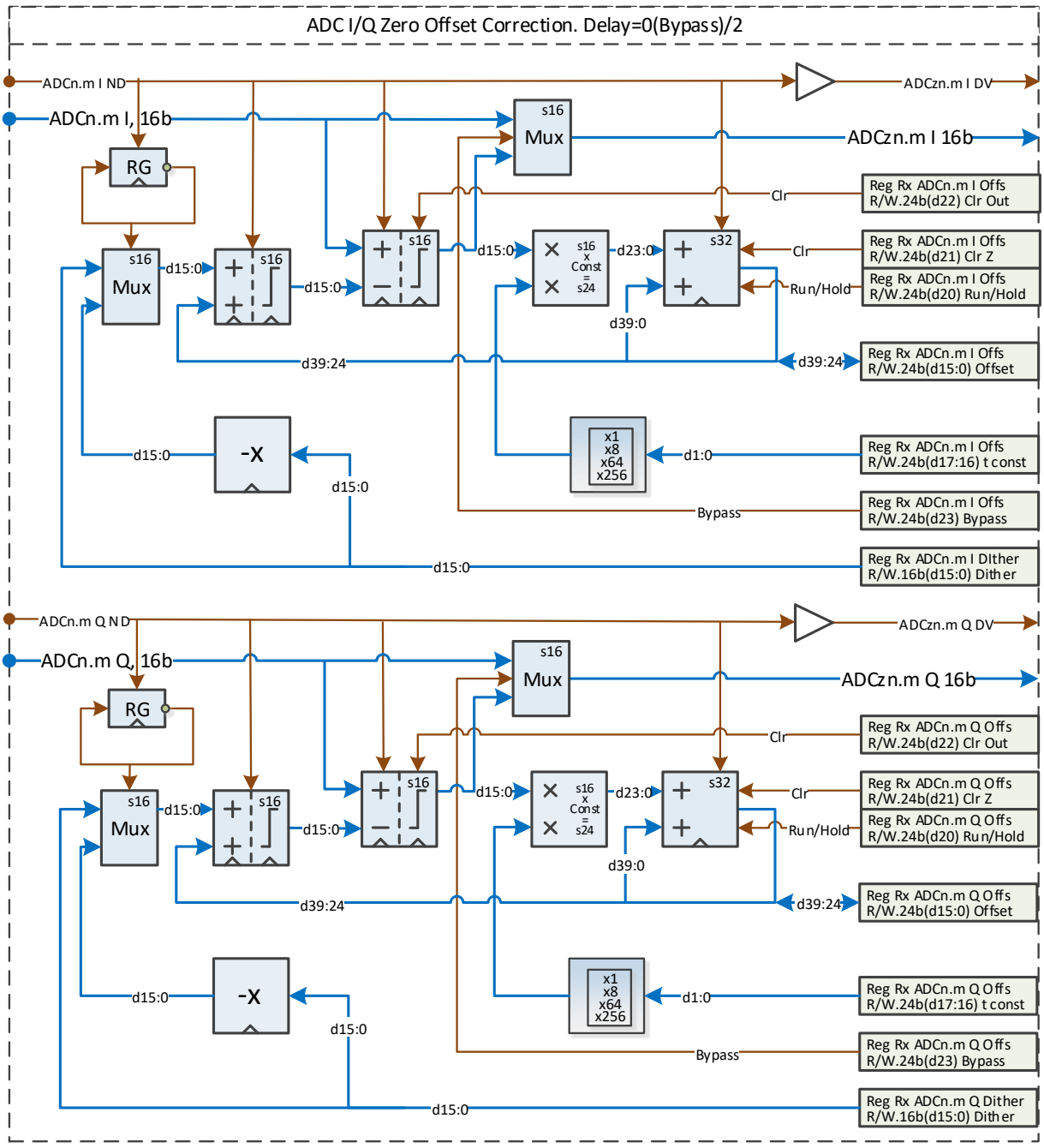


Рисунок 21.65. Модуль коррекции смещения «нулей» квадратурных сигналов от АЦП

Модуль коррекции смещения нулей квадратурных сигналов может конфигурироваться в регистрах управления по командам от модуля DFE_Commander в несколько различных режимов работы:

- Bypass - режим обхода, без обработки данных;

- Clr – режим обнуления выходных данных;
- Hold – режим заморозки приема и выдачи данных с сохранением записанных значений;
- Run – режим с коррекцией нулей по фиксированным значениям смещений сигналов IQ;
- Auto Run – режим коррекции смещения нулей с автоматическим вычислением уровня смещения по каждой квадратурной составляющей.

В режиме Auto Run, который задается единичным значением поля

«(d20) Run/Hold» и нулевым значением поля «(d21) Clr Z» в 24-х битном конфигурационном регистре Reg Rx ADCn.m I(Q) Offs, постоянная времени вычисления смещений зависит от значения двухбитового поля t.const, и определяется по формуле:

τ = 65536/Fsamp, при t.const =0;

τ = 2^18/Fsamp, при t.const =1;

τ = 2^21/Fsamp, при t.const =2;

τ = 2^24/Fsamp, при t.const =3;

где Fsamp означает частоту сэмплирования конкретного канала АЦП в Гц.

Текущие 16-ти битные значения смещений квадратурных сигналов могут быть считаны, зафиксированы или предустановлены в поле “(d15:0) Offset” 24-х битного регистра Reg Rx ADCn.m I(Q) Offs.

21.3.2.2.2.2 Коррекция коэффициентов усиления квадратурных сигналов

Модуль коррекции коэффициентов усиления квадратурных сигналов (Рисунок 21.66) может конфигурироваться в регистрах управления по командам от модуля DFE_Commander в 4 различных режима работы:

- Bypass - режим обхода, без обработки данных;
- Clr – режим обнуления выходных данных;
- Hold – режим заморозки приема и выдачи данных с сохранением записанных значений;
- Run – режим с коррекцией усиления квадратурных каналов по фиксированным значениям, заданным в полях «(d15:0) Gain» 24-х битных регистров Reg Rx ADCn.m I(Q) Gain.

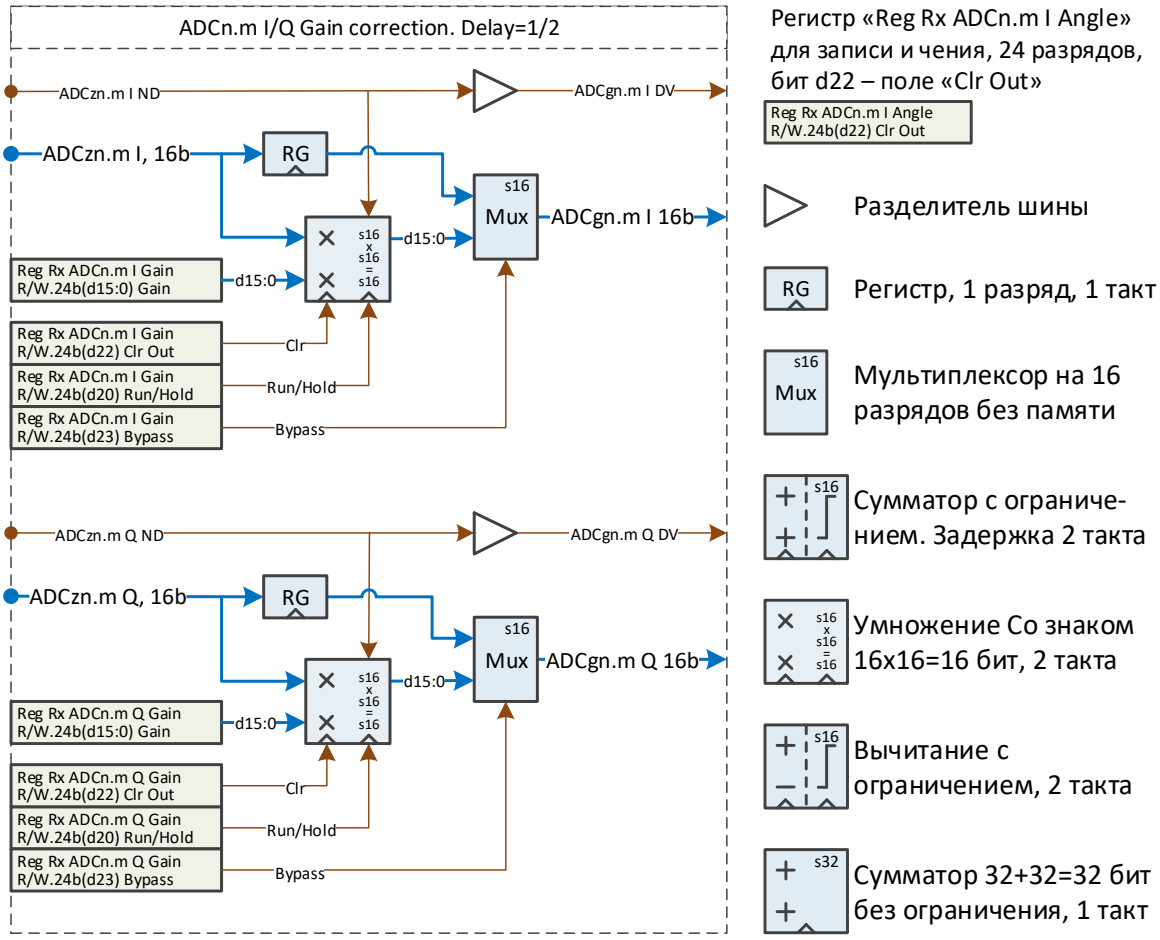


Рисунок 21.66. Модуль коррекции коэффициентов усиления квадратурных сигналов от АЦП

21.3.2.2.2.3 Коррекция неортогональности квадратурных сигналов

Модуль коррекции неортогональности квадратурных сигналов (Рисунок 21.67) может конфигурироваться в регистрах управления по командам от модуля DFE_Commander в 4 различных режима работы:

- Bypass - режим обхода, без обработки данных;
- Clr – режим обнуления выходных данных;
- Hold – режим заморозки приема и выдачи данных с сохранением записанных значений;
- Run – режим с коррекцией неортогональности квадратурных каналов на фиксированный угол (α), значение Sin (α) которого задается в поле «(d15:0) Sin (α)» 24-х битного регистра Reg Rx ADCn.m Angle.

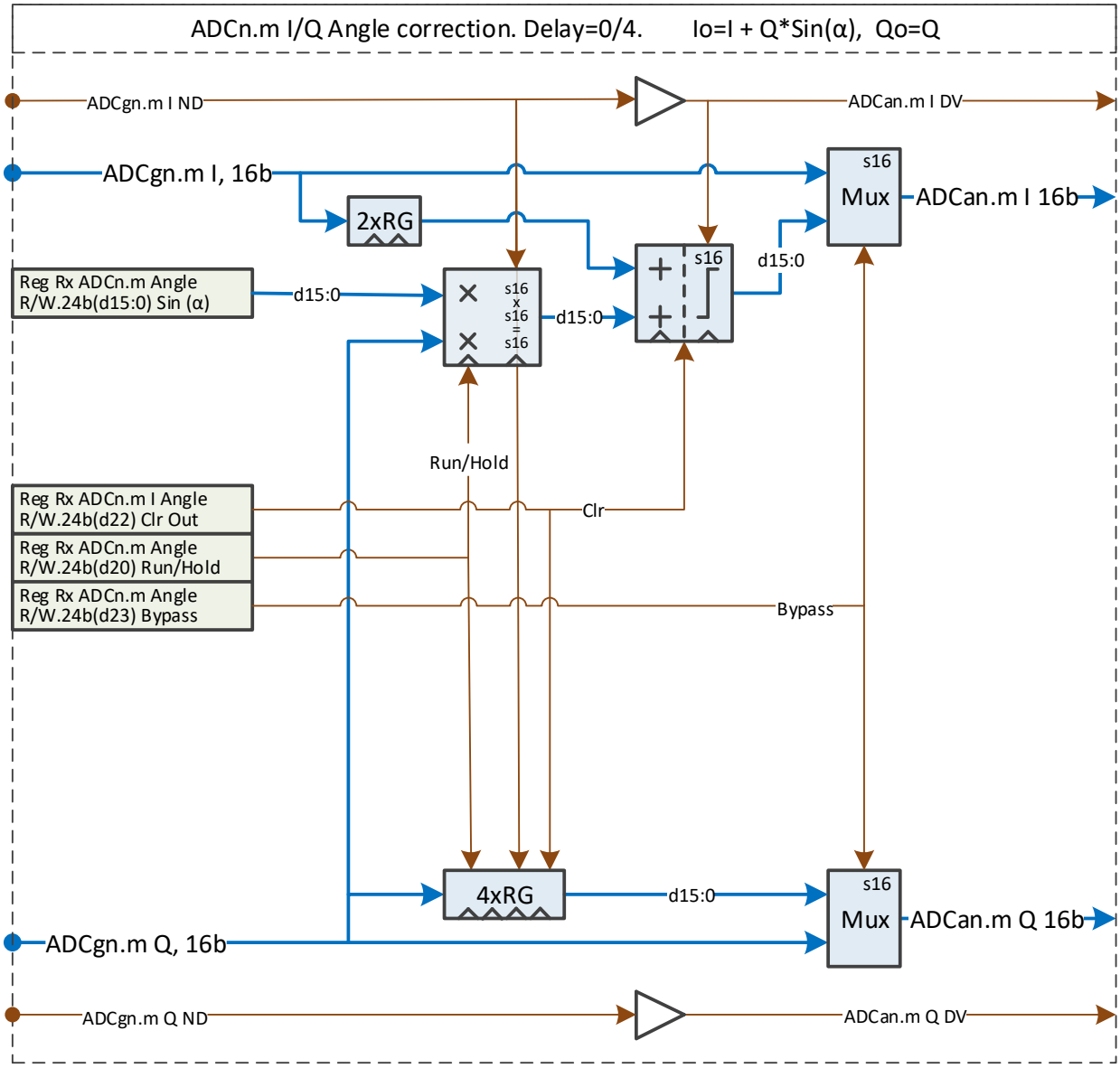


Рисунок 21.67. Модуль коррекции неортогональности квадратурных сигналов от АЦП

21.3.2.2.3 Аттенюаторы

С целью расширения динамического диапазона принимаемых сигналов в модуль приема данных от АЦП блока DFE введен модуль управления цифровыми аттенюаторами радиотракта. Аттенюаторы типично устанавливаются по 2 штуки на каждый канал, один на антенном входе приемного тракта, второй - в цепях преселектора после входного МШУ перед первым смесителем, если используется аналоговое преобразование частоты принимаемого сигнала на промежуточную частоту ПЧ.

Возможно подключение до 2-х последовательно включенных цифровых аттенюаторов в тракте каждого из 8-ми принимаемых каналов, оцифровываемых отдельным каналом АЦП. Один из пары аттенюаторов обеспечивает грубую регулировку усиления, с шагом 6 или 10 дБ, а другой – точную, с шагом 1 дБ. Таким образом, суммарное ослабление сигнала в

приемном канале можно довести до 61 дБ. При необходимости грубый аттенюатор может отсутствовать.

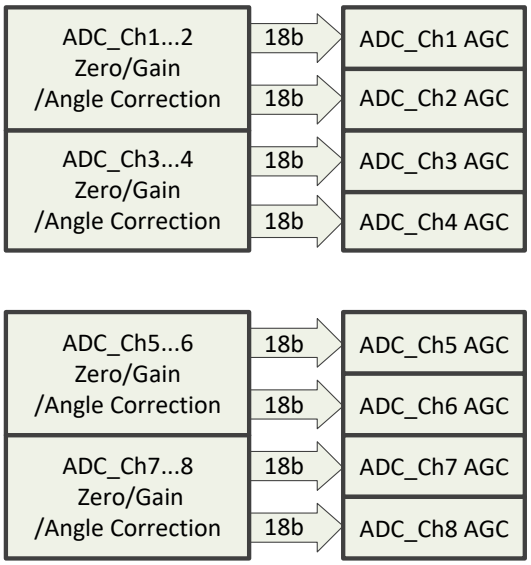


Рисунок 21.68

Аттенюаторы управляются по интерфейсу DFE SPI4 через 8-ми битные команды. 2 старших бита кода команды определяют грубую аттенюацию с шагами 0/10/20/30 дБ или 0/6/12/18 дБ, 5 следующих битов задают аттенюацию 0-31 дБ с шагом 1 дБ. Младший бит всегда равен 0.

Все 8 пар аттенюаторов для 8-ми независимых каналов приема должны подключаться к последовательному интерфейсу DFE SPI4, и разделяются по адресам сигнала /DFE_SPI4_CS1…4 (низкий активный уровень).

К одному интерфейсу SPI4 могут быть подключены до 4-х внешних устройств с адресами CS 1, 2, 4, 8, при выборе которых одна из 4-х линий /CS1…/CS4 опускается в нулевой уровень, сигнализируя соответствующему внешнему устройству о передаче ему данных по шине DFE_SPI4_MOSI синхронно с тактовым сигналом DFE_SPI4_CLK. При этом аттенюаторам с адресами CS 1, 2, 4, 8 соответствуют обрабатываемые каналы АЦП ADC_Ch1, ADC_Ch3, ADC_Ch5, ADC_Ch7.

При необходимости, к интерфейсам DFE SPI1…4 могут быть подключены более 4-х внешних устройств, для чего к выводам /DFE_SPI4_CS1…4 необходимо подключить дешифратор 4 в16, выходы которого, кроме позиции 16 (когда сигналы DFE_SPI4_CS1…4 = 1), могут адресовать до 15-ти внешних устройств.

Таким образом, при необходимости использования более 4-х пар аттенюаторов, все 8 пар аттенюаторов подключаются через дешифратор и имеют адреса 1, 3, 2, 5, 4, 6, 8, 7 соответствующие активным каналам ADC_Ch1…ADC_Ch8.

Для подключения аттенюаторов с параллельным заданием уровнем аттенюации, можно использовать различные расширители портов SPI, преобразующие последовательные команды по 8 бит в параллельную 8-ми битную шину управления грубым и точным аттенюаторами.

Если для приема сигналов используются квадратурные сигналы IQ от 2-х АЦП одновременно, то блок управления аттенюаторами необходимо устанавливать в режим квадратурной обработки, при котором аттенюаторы с номерами 1, 2, 4, 8 управляют 4-мя независимыми каналами приема, каждый из которых оцифровывается парами АЦП ADC_Ch1+ADC_Ch2, ADC_Ch3+ADC_Ch4, ADC_Ch5+ADC_Ch6, ADC_Ch7+ADC_Ch8 соответственно.

Управление аттенюаторами происходит через регистры “Attenuator_ch_settings”, “Attenuator_ch_data0”, “Attenuator_ch_data1”, “Attenuator_ch_uplevel” доступные только через интерфейс APB

21.3.2.2.4 Модуль Digital Down Converter

С выходов модулей управления аттенюатором потоки данных поступают в модули DDC, выполняющие функции переноса спектра сигналов на нулевую или малую промежуточную частоту (при необходимости) с формированием квадратурных IQ составляющих сигнала, их последующей фильтрации, децимации, ресэмплирования и записи выходных потоков в память DSP через буферы FIFO посредством каналов DMA.

Структурная схема одного кластера из 4-х модулей DDC (DDUC) и одного канала DUC, управляемых от одного из 4-х потоков DFE Commander, приведена на Рисунок 21.69.

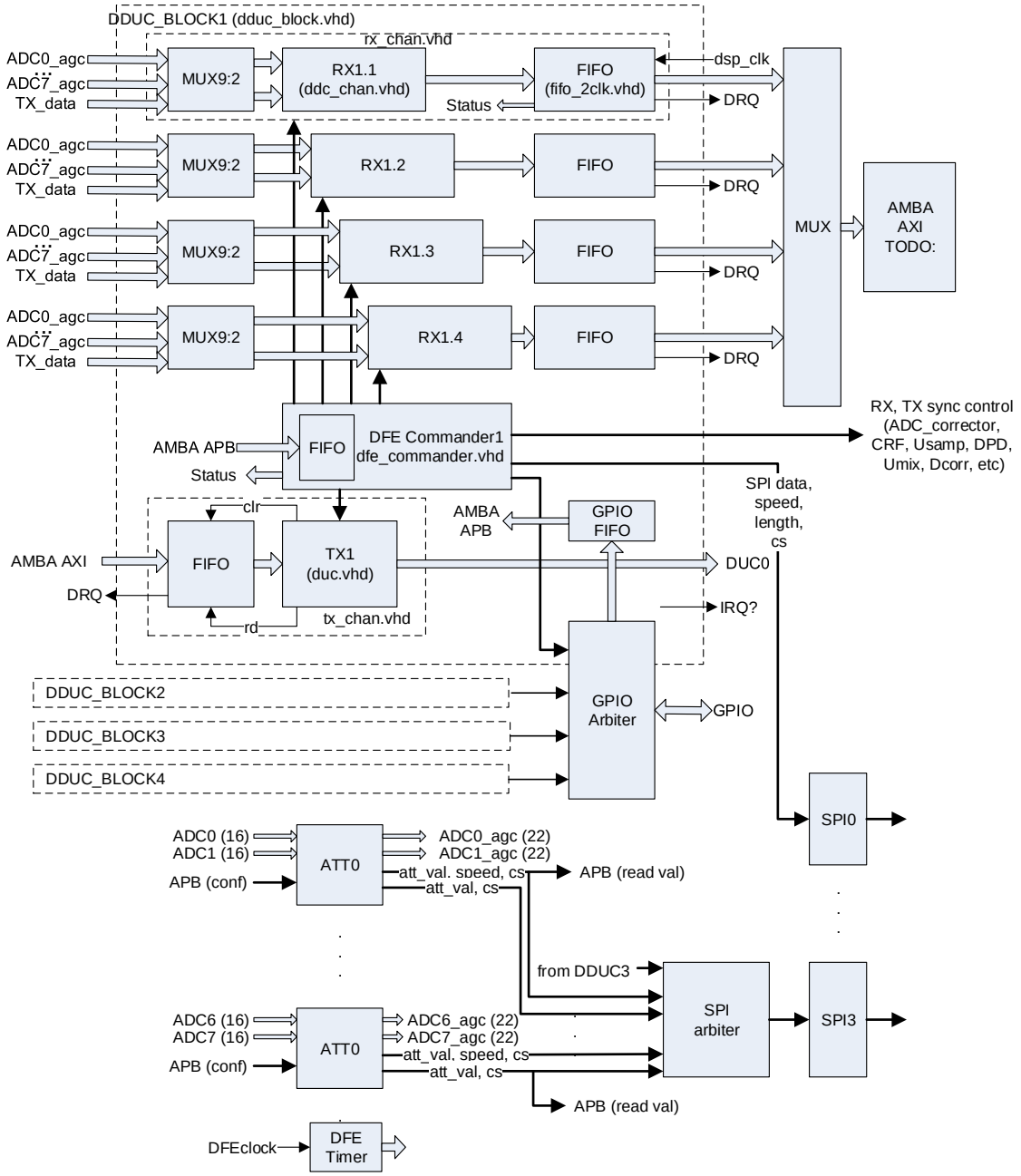


Рисунок 21.69. Структурная схема 4-х модулей DDC и одного канала DUC, управляемых от одного из потоков DFE Commander

Структурная схема одного модуля DDC приведена на Рисунок 21.70.

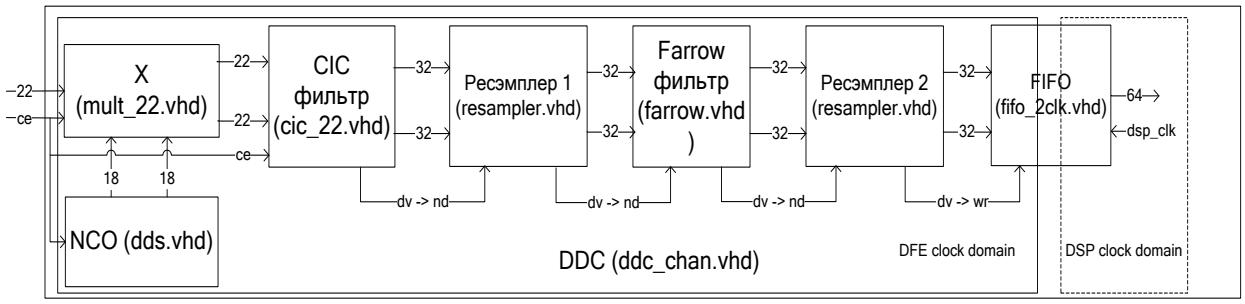


Рисунок 21.70. Структура ddc_chan

На входе модуля DDC стоит смеситель с сигналом цифрового гетеродина. Далее сигнал децимируется с помощью CIC-фильтра. После CIC-фильтра сигнал проходит через фильтр «ресэмплер 1», основная задача которого коррекция АЧХ после CIC фильтра и фильтрация помеховых сигналов вне полосы приема. Далее сигнал проходит через фильтр Фарроу, который используется для передискретизации сигнала и подстройки фазы, и затем, сигнал проходит еще через один фильтр «ресэмплер 2», основная задача которого выполнение согласованной фильтрации.

Управление модулем DDC осуществляется через регистр «Control reg»

21.3.2.2.4.1 Цифровой гетеродин (DDS)

Цифровой гетеродин имеет следующие параметры – частота, фаза и изменение частоты. Частота задается в виде 32-х битного числа, задающего приращение фазы, вычисленной по формуле 2^{32} * F / Fclk, где F – частота гетеродина в Гц, а Fclk частота оцифровки сигнала на входе смесителя. Фаза задается в виде 32-х битного числа по формуле 2^{32} * φ / 360, где φ фаза в градусах. Изменение частоты задается также в виде 32-х битного слова и вычисляется по следующей формуле 2^{32} * ΔF / Fclk, где ΔF скорость изменения частоты заданной в Гц/с. На выходе цифрового гетеродина формируется сигнал разрядностью 18 бит.

Управление гетеродином осуществляется через регистры «DDS RX FREQ», «DDS RX PHASE» и «DDS RX FREQ SHIFT».

21.3.2.2.4.2 Фильтр CIC

CIC-фильтр состоит из 5 секций и обеспечивает децимацию сигнала в 1-16384 раза. Его структура показана на Рисунок 21.71.

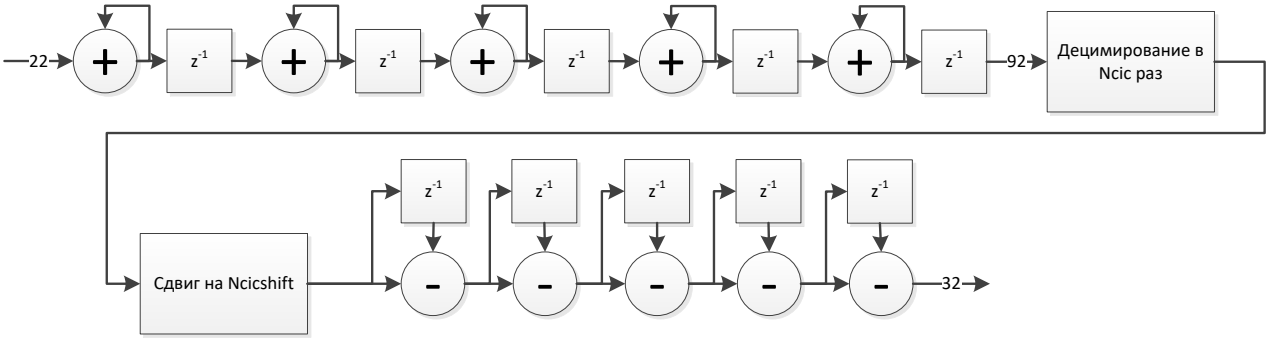


Рисунок 21.71. Структура CIC-фильтра в DDC

На входе фильтра сигнал имеет разрядность 22 бита, на выходе 32 бита. Разрядность внутренних секций интегрирования составляет 92 бита. Усиление фильтра составляет:

$GAINcic = Ndec^5 * 2^{(-Ncicshift)}$, где Ndec – коэффициент децимации, а Nshift количество сдвигов, и может быть в диапазоне 0-63.

При программировании необходимо задавать коэффициент децимации как Ncic-1. Если коэффициент децимации равен 1, то модуль CIC фильтра работает на проход, сдвиг на Ncicshift разрядов в этом случае не выполняется, и в качестве выходного сигнала берется входной, расширенный до 32-х бит и умноженный на 8.

Управление фильтром осуществляется через регистры «CIC RX» и «CIC RX gain».

21.3.2.2.4.3 Ресемплеры

Модули «Ресэмплер 1» и «Ресэмплер 2» представляют из себя обычные FIR фильтры с интерполяцией в 1-128 раза и децимацией в 1-128 раза. Максимальное количество коэффициентов фильтра «Ресемплер1» 512 бит, фильтра «Ресемплер 2» 256 бит (для передатчика максимальное количество коэффициентов фильтра «Ресемплер1» 256 бит, фильтра «Ресемплер 2» 512 бит). Коэффициенты фильтра 18 разрядные. Входные и выходные данные имеют разрядность 32 бита. В каждом канале I и Q каждый фильтр имеет по 8 умножителей 32x18, что определяет максимальную производительность фильтра. То есть для фильтра длиной 512 необходимо 64 такта для получения выхода, в случае если коэффициенты интерполяции и децимации равны 1.

Загрузка коэффициентов фильтра осуществляется в следующем порядке. Если коэффициент интерполяции равен 1, то коэффициенты фильтра загружаются последовательно. Длина фильтра должна быть кратна 8n, где n – коэффициент интерполяции. Если длина фильтра не кратна 8n, то к фильтру добавляются нулевые коэффициенты в конец, до длины кратной 8n. Если коэффициент интерполяции фильтра n больше 1, то порядок загрузки коэффициентов фильтра следующий. Сначала загружаются первые 8n коэффициентов: C0, Cn, C2n, C3n, C4n, C5n, C6n, C7n, C1, Cn+1, C2n+1, C3n+1, …, C7n+1, C2, Cn+2, C2n+2, … Cn-1, C2n-1,….,C8n-1. Затем загружаются следующие 8n коэффициентов в аналогичном порядке C8n, C9n, …, C15n, C8n+1, C9n+1, … C15n+1, .., C16n-1. И так далее все последующие блоки коэффициентов.

21.3.2.2.4.3.1 Порядок загрузки коэффициентов resampler

Перед загрузкой необходимо снять сигнал nclr (командой 5 dfe_commander). Перед началом загрузки таблицы надо подать 0x800000 чтобы сбросить указатель записи в 0.

Порядок загрузки в псевдокоде:

```
reg_load(0x800000);
for(int k=0;  k<blocks-1; K++)
  for(int  i=0; i<M-1; i++)
    for(int j=0; j<BUF_NUM; j++)
      if(k*block_size+i+j*M < len)
   reg_load(coef(k*block_size+i+j*M));
          else
       reg_load(0);
```

Пояснения:

BUF_NUM равен 4 для ресемплеров с максимальной длиной в 256 и 8 для ресемплеров с максимальной длиной 512

M – коэффициент интерполяции

len – длина таблицы

block_size = M*BUF_NUM

blocks = len/block_size, округление вверх до ближайшего целого

reg_load – функция которая загружает данные в регистр.

coef – массив с таблицей коэфициентов ресемплера.

Для режима bypass ресемплера надо установить длину таблицы 2, коэффициенты M и D равными 1. Загрузить таблицу: первый коэффициент 1FFFF и следующие 3 (для BUF_NUM =4) или 7 (для BUF_NUM = 8) коэффициентов нулевые.

Управление ресемплерами осуществляется через регистры “Resampler RX coefs”, “Resampler RX length”, “Resampler RX m_factor” и “Resampler RX d_factor”.

21.3.2.2.4.4 Фильтр Фэрроу

Фильтр Фарроу используется для передискретизации сигнала, а также для задержки сигнала на определенную величину. Фильтр позволяет децимировать сигнал в диапазоне 1..16-260 и интерполировать в диапазоне 1..260. Вычисление промежуточных точек выполняется на основе интерполяционного полинома Лагранжа третьей степени. По следующей формуле:

$$Sout = a_0 + \mu * \left(a_1 + \mu * (a_2 + \mu * a_3)\right), \text{где}$$

$$a_3 = \frac{1}{6}S(t_0) - \frac{1}{2}S(t_{-1}) + \frac{1}{2}S(t_{-2}) - \frac{1}{6}S(t_{-3})$$

$$a_2 = \frac{1}{2}S(t_0) - S(t_{-1}) + \frac{1}{2}S(t_{-3})$$

$$a_1 = \frac{1}{3}S(t_0) + \frac{1}{2}S(t_{-1}) - S(t_{-2}) + \frac{1}{6}S(t_{-3})$$

$$a_0 = S(t_{-1})$$

S(t) – входные отсчеты, а μ – дробная фаза в диапазоне -1..0.

Внутренняя точность вычисления коэффициентов a0-a3 составляет 36 бит.

Для работы фильтра необходимо задать коэффициент передискретизации сигнала в формате 4Q60, при этом, если коэффициент больше 1, то выходной сигнал децимируется, а если меньше единицы, то интерполируется. Производительность фильтра такова, что он может выдавать данные каждый такт, таким образом, частота поступления входных данных должна быть такой, чтобы в случае интерполяции фильтр успел выдать все промежуточные значения между выборками.

Управление фильтром осуществляется через регистры “Farrow low RX” и “Farrow high RX”.

21.3.2.2.5 Канал bypass

Модуля ADC_DDC_Bypass, предназначенного для организации считывания данных от АЦП непосредственно в память кластера DSP без предварительной обработки. Данные для этого канала снимаются сразу после модулей Zero/Gain/Angle correction.

Управление режимом байпас осуществляется через регистр «ADC DDC Bypass»

21.3.2.3 Передатчик

21.3.2.3.1 Криптомодуль

Предотвращает нежелательную передачу данных. Если передаваемые данные не соответствуют кодовому слову любая передача данных прекращается.

Варианты кодового слова:

0xC4D7	в данных разрешено только нечетное число единиц
0x3B23	в данных разрешено попеременно четное и нечетное число единиц

0xEB23	защита отключена, разрешена передача любых данных
Иные кодовые слова	передача любых данных запрещена

21.3.2.3.2 Фифо передатчика

Модули FIFO DSP-DUC1-4 обеспечивают передачу асинхронных потоков данных DMA от DSP синхронно к блоку DFE. Работа модулей FIFO синхронизируется в тактовом домене DFE_Clk. Кроссинхронизация между блоками DSP и DFE производится на стыке шины AXI и DMA.

Предельные скорости входных потоков данных, передаваемых из DSP в блок DFE, а также максимальные размеры буферов ОЗУ для блоков данных, приведены на структурной схеме модуля TX 1 to 4 channels Digital Up Converter.

21.3.2.3.3 Digital Up Converter

С выходов модулей FIFO DSP-DUC1-4 4 потока данных поступают на вход модулей цифровых передающих конверторов DUC1-4, обеспечивающих интерполяцию, ресэмплирование, цифровую фильтрацию и перенос спектра каналов на промежуточные частоты для дальнейшего их суммирования-объединения с другими каналами.

Структура модуля DUC аналогична DDC (Рисунок 21.72), но блоки располагаются в обратном порядке. Сначала сигнал проходит через «Ресэмплер 1», затем через фильтр Фарроу для передискретизации, затем идет «Ресэмплер 2», и далее CIC-фильтр.

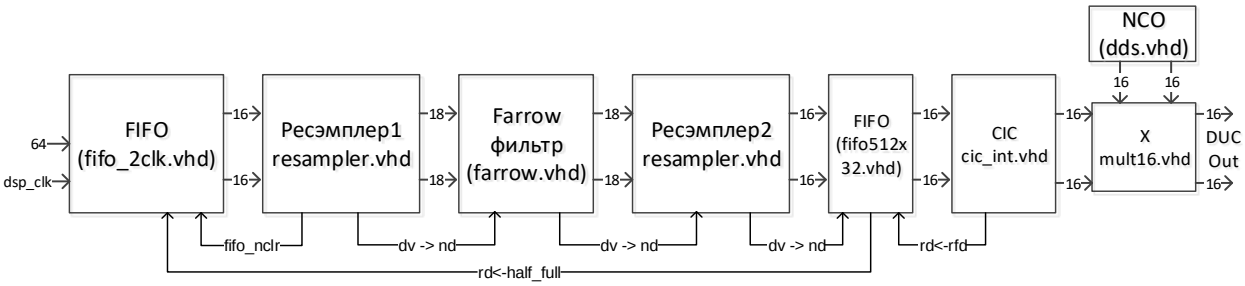


Рисунок 21.72. Структура модуля DUC

Модули «Ресэмплер 1» и «Ресэмплер 2» полностью аналогичны модулям, используемым в DDC, за тем исключением, что сигнал на входе и выходе имеет разрядность 16 бит и используется другое количество коэффициентов (см. Раздел «Ресемплеры»).

Фильтр Фарроу также аналогичен фильтру, применяемому в DDC, за исключением того, что входные и выходные данные имеют разрядность 16 бит.

Управление этим блоком аналогично управлению блоком DDC, но в обратном порядке. Управление каждым DUC происходит через соответствующий DFE commander, используются следующие регистры:

- ресемплеры – “Resampler TX coefs”, “Resampler TX length”, “Resampler TX m_factor” и “Resampler TX d_factor”
- фильтр Фэрроу – “Farrow low TX” и “Farrow high TX”
- фильтр CIC – “CIC TX” и “CIC TX gain”
- Цифровой гетеродин – “DDS TX FREQ”, “DDS TX PHASE” и “DDS TX FREQ SHIFT”
- DUC – “Control reg TX”

21.3.2.3.3.1 CIC фильтр

CIC-фильтр имеет 5 секций и позволяет интерполировать сигнал в 1-16384 раза. Структура фильтра показа на Рисунок 21.73.

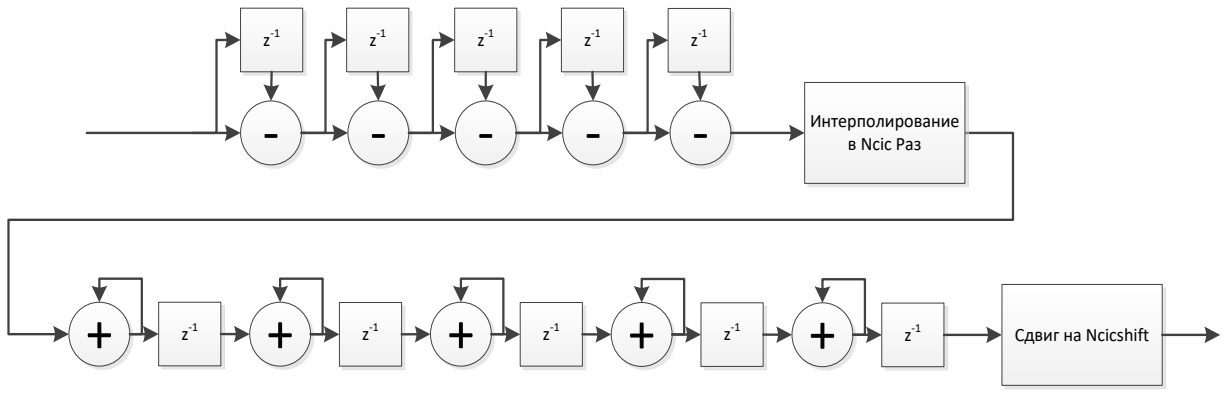


Рисунок 21.73. Структура CIC-фильтра в составе TX DUC

На входе и выходе фильтр имеет разрядность 16 бит. Коэффициент усиления фильтра определяется следующей формулой:

$GAINcic = Ndec^4 * 2^{(-Ncicshift)}$, где Ndec – коэффициент децимации, а Nshift количество сдвигов, и может быть в диапазоне 0-63.

Цифровой гетеродин аналогичен модулю, используемому в блоке DDC.

21.3.2.3.4 TX channel summators

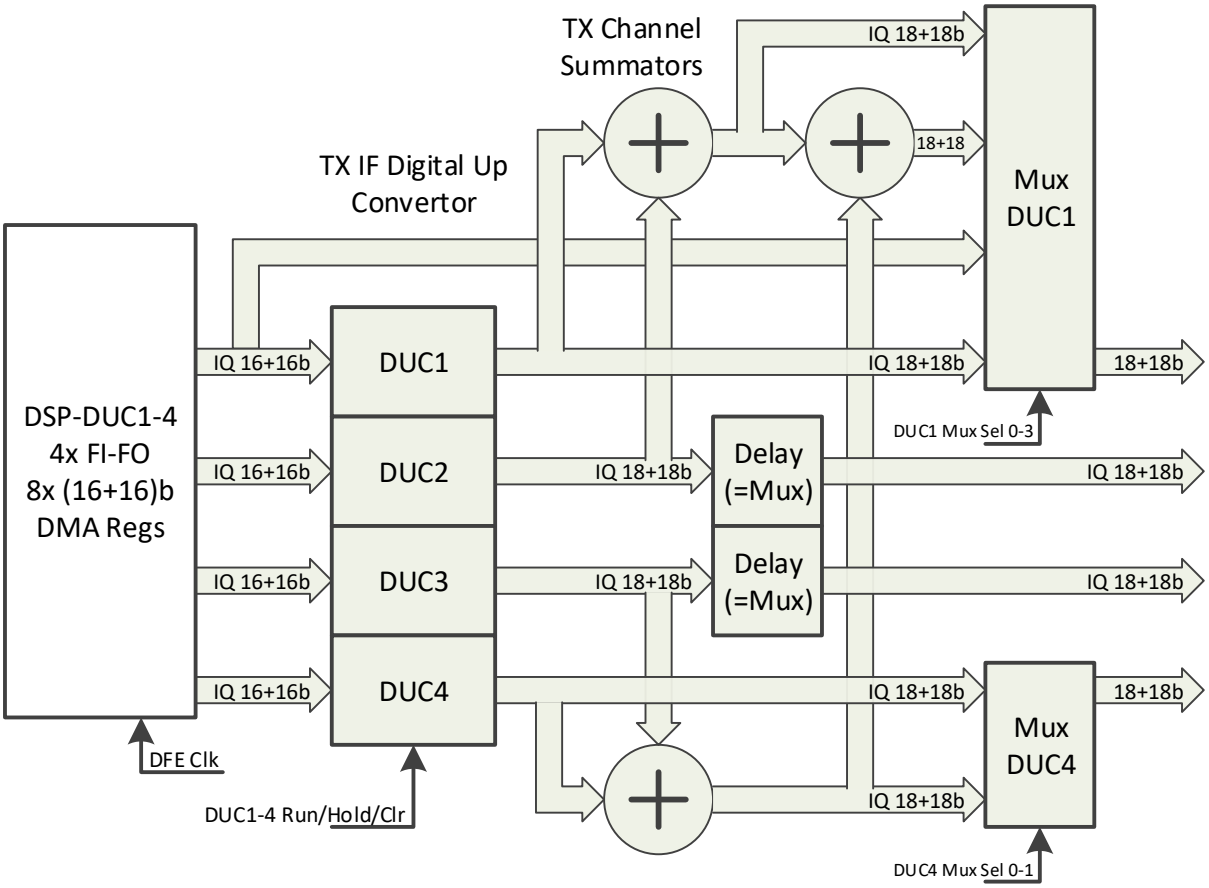


Рисунок 21.74

С выходов 4-х каналов DUC потоки данных поступают на модуль сумматоров и мультиплексоров TX Channel Summators, позволяющий, при необходимости, суммировать несколько сформированных частотных каналов на промежуточной частоте с целью дальнейшей передачи одного суммарного многоканального сигнала посредством одного передатчика.

Управляется данная схема через регистр «Tx_Sum» доступный через интерфейс APB.

21.3.2.3.5 Crest Factor Reduction

С выхода модуля TX Channel Summators сигналы каналов промежуточной частоты поступают на модули Crest Factor Reduction (CFR) снижения пик-фактора передаваемого сигнала, функциональная схема Simulink модели которого приведена на Рисунок 21.75.

Принцип работы модуля CFR основан на нахождении пиковых всплесков квадратурного сигнала и формирование узкополосного комплексного противоимпульса, вычитание которого из сигнала позволяет значительно снизить значение пиковых всплесков без значимого ущерба для повышения уровня внеполосных искажений передаваемого сигнала.

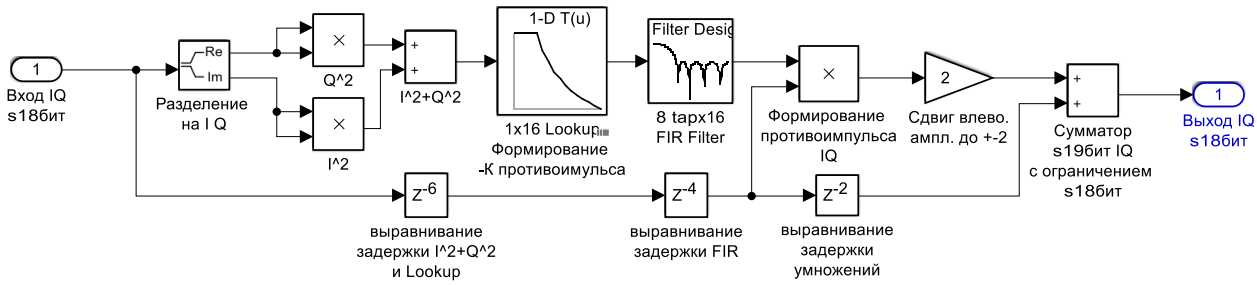


Рисунок 21.75. Функциональная схема Simulink модели модуля Crest Factor Reduction

Предложенное решение позволяет снизить пик-фактор сложных сигналов на 3-5 дБ, что позволяет в 2-3 раза повысить уровень излучения средней мощности передатчика без внесения значительных искажений в передаваемый сигнал.

В модуле CFR используется одномерная таблица, содержащая 16 16-тиразрядных коэффициентов, и блок симметричного 15 tap FIR фильтра. Таблица коэффициентов разделена на две части: пока одна часть используется в работе доступна запись во вторую часть. Все коэффициенты таблицы и фильтра задаются программно через DFE Commander.

Если нет необходимости использовать модуль CFR, то в любых из 4-х каналов TX можно установить режим обхода - Bypass, при котором сигнал минует модули CFR.

Управление осуществляется через регистры “TX CFR FIR”, “TX CFR Controlu” и “TX CFR lookup”

21.3.2.3.6 Digital Predistortion

Одним из наиболее эффективных методов снижения внеполосных излучений передатчиков при передаче сложных сигналов с амплитудной составляющей модуляции и повышенным пик-фактором является метод формирования адаптивных предыскажений передаваемого сигнала, которые будут скомпенсированы нелинейной передаточной характеристикой усилителя мощности передатчика, в результате чего сигнал будет излучен с низким уровнем нелинейных искажений.

С целью повышения линейности тракта передачи радиостанции в канале формирования передаваемого сигнала блока DFE используется модуль Digital Predistortion (DPD), функциональная схема Simulink модели которого показана на Рисунок 21.76.

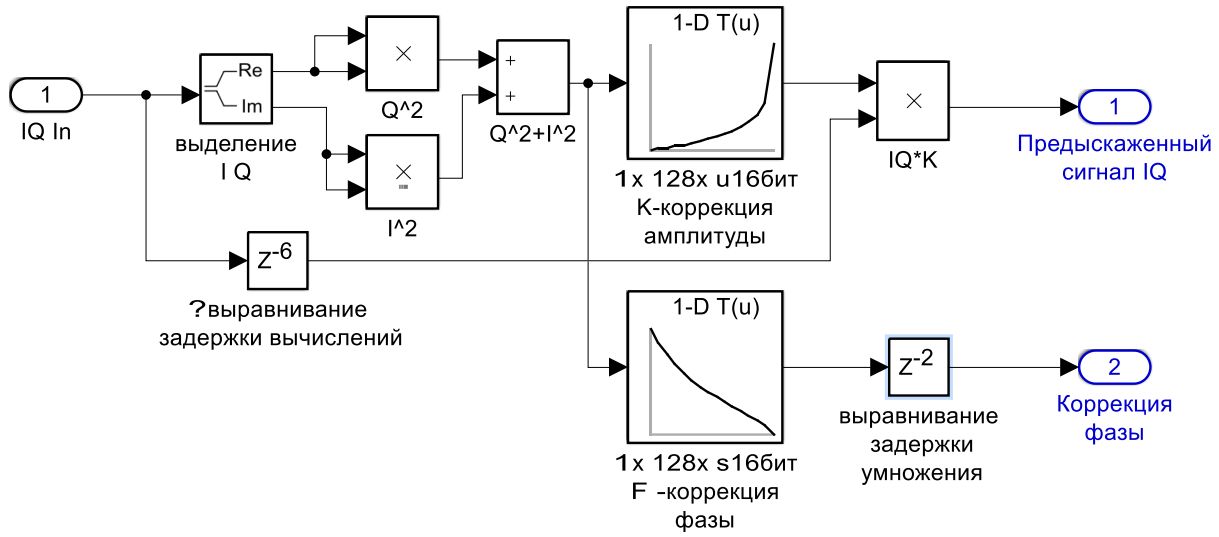


Рисунок 21.76. Функциональная схема Simulink модели модуля DPD

Модуль DPD при работе использует 2 сменных пары одномерных таблицы из 64-х 16-тиразрядных коэффициентов каждая. Первая таблица содержит предыскаженные коэффициенты передачи тракта в зависимости от амплитуды входного сигнала. Вторая таблица содержит корректирующие значения фазовых сдвигов сигнала, которые применяются для коррекции в следующем модуле цифрового гетеродина и переноса частоты TX RF/IF Up Mixer. Все коэффициенты таблиц загружаются посредством DFE Commander в нерабочую версию таблиц, а по команде переключения рабочая и нерабочая пары таблиц меняются местами.

С целью облегчения вычисления характеристик амплитудных и фазовых предыскажений в блоке DFE предусмотрена петля обратной связи для сигналов на входе модуля DPD, которые могут быть считаны в память DSP кластера через свободные каналы DDC. Передаваемый сигнал с выхода передатчика так же может быть считан одним и свободных каналов DDC с целью сравнения и вычисления корректирующих коэффициентов таблиц.

Совместное применение методов CFR и DPD позволяет помимо снижения уровня внеполосных искажений значительно повысить уровень излучения средней мощности передаваемого сигнала.

Модуль DPD так же имеет программируемый режим обхода - Bypass.

Управление осуществляется через регистры “TX Digital Predistortion Amp”, “TX Digital Predistortion Ph” и “TX Digital Predistortion Control” с доступом через DFE Commander.

21.3.2.3.7 Модуль Up Mixer

Модуль TX IF/IF Up Mixer цифрового гетеродина и квадратурного смесителя предназначен для переноса спектра сигнала промежуточной частоты на требуемую частоту передачи сигнала, а также для точной подстройки его частоты и фазы. Модуль обеспечивает следующие режимы работы:

- частотный перенос спектра квадратурных сигналов с высокой точностью (аккумулятор фазы = 32 бит);
- точное задание и коррекцию фазы передаваемых сигналов;
- скачкообразный программируемый перенос спектра сигнала для режима ППРЧ;
- программируемый плавный линейный перенос спектра сигнала для режима ЛЧМ;
- комбинированный режим ППРЧ + ЛЧМ + ФМ;
- режим обхода – Bypass.

Управление осуществляется через регистры “IQ Up DDS TX FREQ”, “IQ Up DDS TX PHASE”, “IQ Up DDS TX FREQ SHIFT”, “IQ Up Mixer Control”, “IQ UpSampling FIR Coefs” и “IQ UpSampling” с доступом через DFE Commander

21.3.2.3.8 Zero/Gain/Angle correction

Модули коррекции неортогональности, коэффициентов усиления и нулей передаваемых квадратурных сигналов

Их работа в обратном порядке аналогична работе таких блоков в тракте приема данных от АЦП, за исключением отсутствия схемы автоматического вычисления смещения нулей выдаваемых квадратурных сигналов. В данном случае коррекция смещения нулей производится по фиксированным значениям смещений, задаваемых в полях “(d15:0) Offset” 24-х разрядных регистров Reg Tx DUCn I(Q) Offs.

Управление осуществляется через регистры «Angle Correction TX», «Gain Correction I TX», “Gain Correction Q TX”, “Zero offset correction I TX” и “Zero offset correction Q TX” с доступом через DFE Commander.

21.3.2.3.9 Интерфейс к ЦАП

Модуль интерфейсов ЦАП предназначен для преобразования потока выходных данных блока DFE к формату, удовлетворяющему требованиям используемых ЦАП.

Спецификация интерфейса:

- возможность подключения одной, двух или трех ИМС ЦАП;
- типы используемых интерфейсов LVDS/CMOS SDR/DDR, JESD204B;
- максимальная разрядность интерфейсов:

- CMOS - 16 бит;

- LVDS - 16 пар;

- независимые входы синхронизации для интерфейсов CMOS и LVDS.

Интерфейс модуля схематично показан на Рисунок 21.77.

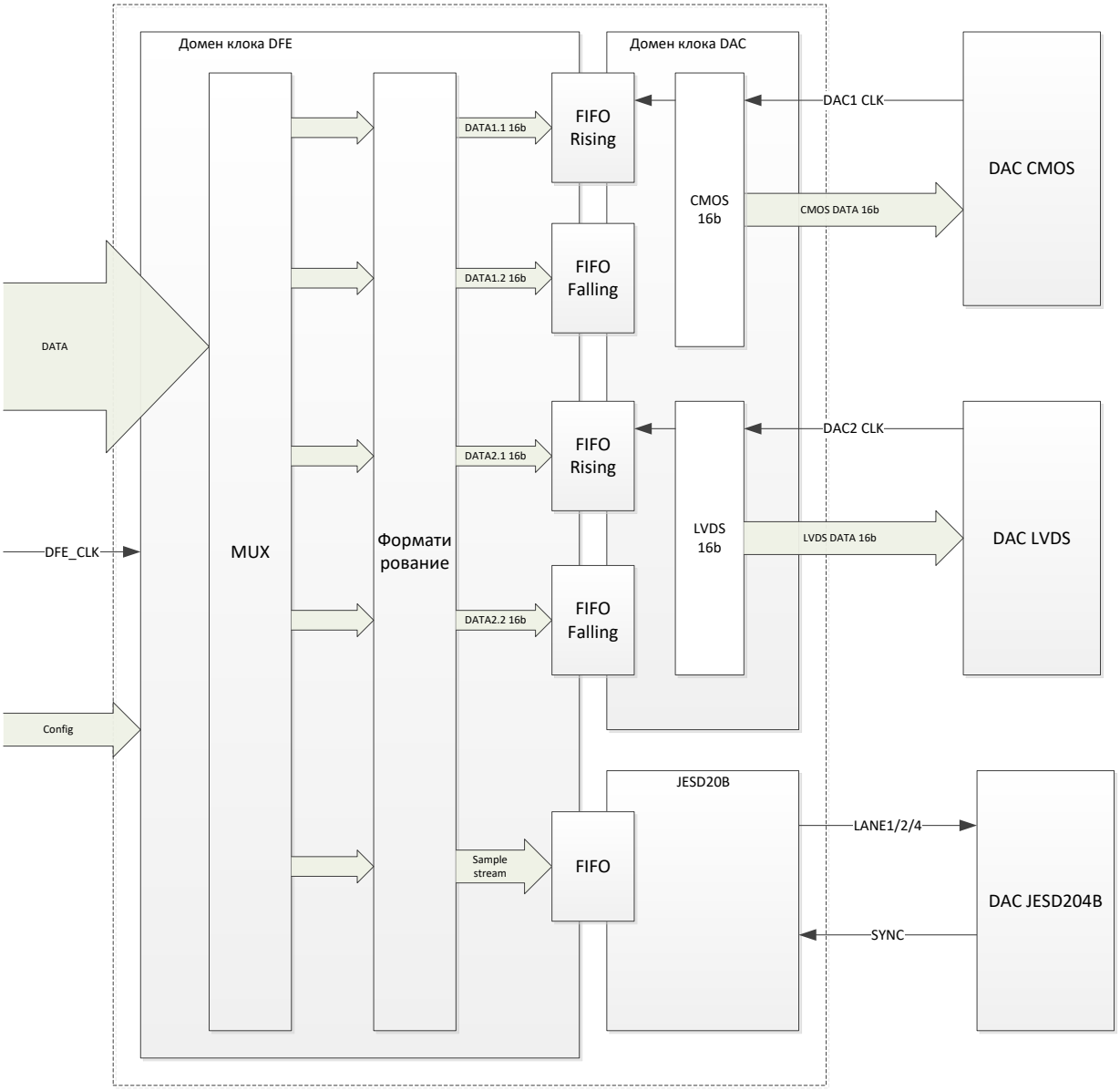


Рисунок 21.77. Функциональная схема модуля интерфейсов ЦАП

21.3.3 Встроенные DMA

Для передачи данных между каналами DFE и внешней памятью используются 9 4-канальных DMA:

1. Два DMA для блока DFE_ALT:

   - Alt_dma_rx обслуживает потоки:

- RX_CH1 – канал 0
- RX_CH2 – канал 1
- RX_CH3 – канал 2
- RX_CH4 – канал 3

- Alt_dma_mix обслуживает потоки:
  - RX_CH5 – канал 0
  - RX_CH6 – канал 1
  - TX_CH1 – канал 2
  - TX_CH2 – канал 3

2. Пять DMA для блока DFE_NELS:

- Nels_dma_ddc0 обслуживает потоки:
  - DDC00 – канал 0
  - DDC01 – канал 1
  - DDC02 – канал 2
  - DDC03 – канал 3
- Nels_dma_ddc1 обслуживает потоки:
  - DDC10 – канал 0
  - DDC11 – канал 1
  - DDC12 – канал 2
  - DDC13 – канал 3
- Nels_dma_ddc2 обслуживает потоки:
  - DDC20 – канал 0
  - DDC21 – канал 1
  - DDC22 – канал 2
  - DDC23 – канал 3

- Nels_dma_ddc3 обслуживает потоки:
  - DDC30 – канал 0
  - DDC31 – канал 1
  - DDC32 – канал 2
  - DDC33 – канал 3
- Nels_dma_adcbp обслуживает один поток ADCBP – канал 0
- Nels_dma_duc обслуживает потоки:
  - DUC0 – канал 0
  - DUC1 – канал 1
  - DUC2 – канал 2
  - DUC3 – канал 3
- Nels_dma_cmd обслуживает потоки:
  - CMD0 – канал 0
  - CMD1 – канал 1
  - CMD2 – канал 2
  - CMD3 – канал 3

Базовые адреса можно найти в документе Главе 6 «Подсистема SDR».

Все DMA, за исключением Nels_dma_cmd, обеспечивают 64-ю передачу данных. Nels_dma_cmd передает данные по 32 бита.

Каналы за один цикл занятия коммутатора передают пачку данных. Размер пачки задается полем WN в регистре CSR соответствующего канала DMA и определяется системными требованиями по передаче данных.

Канал DMA формирует прерывание:

- при единичном состоянии бита DONE;
- при единичном состоянии битов END.

Обнуление битов DONE и END (и снятие соответствующего прерывания) выполняется посредством чтения содержимого регистра CSR или записью в них нуля CPU.

Распределение запросов на прерывание обозначено в Главе 3 «Системная организация микросхемы».

Для управления работой каналы DMA портов содержат следующие регистры:

- Регистр управления и состояния CSR*;
- регистр индекса (физический адрес памяти) IR*;
- регистр начального адреса блока параметров DMA передачи для самоинициализации CP*;
- псевдорегистр управления состоянием бита RUN* регистра CSR.

Состоянием разряда 0 регистра CSR можно управлять, используя адрес псевдорегистра

RUN. При этом остальные разряды этого регистра не изменяются. Эта процедура может быть использована для временной приостановки канала DMA (если не используется процедуры самоинициализации). Если используются процедуры самоинициализации, то канал DMA можно остановить, установив бит RUN=0 в требуемом массиве параметров самоинициализации. При чтении по адресу псевдорегистра RUN считывается содержимое регистра CSR без сброса битов END и DONE.

21.3.3.1 Процедура самоинициализации

Каналы DMA могут выполнять процедуру самоинициализации (выполнение цепочки передач DMA).

Для выполнения самоинициализации в каналах имеется 64-разрядный регистр CP, в котором хранится начальный адрес блока параметров очередного DMA обмена. Младший (нулевой разряд) регистра CP используется для старта режима самоинициализации. Эти параметры при самоинициализации аппаратно загружаются в 64-разрядном (32-разрядном для Nels_dma_cmd) формате в соответствующие регистры канала DMA. Процедура этой загрузки ничем не отличается от обычного DMA обмена. Блок параметров может размещаться в любой памяти.

Параметры для самоинициализации размещаются в памяти в трех последовательных 64-разрядных словах, следующим образом.

{unus[63:32],wcx[15:0],csr[15:0]}, cp[63:0], ir[63:0]

Если необходимо продолжить цепочку команд, то необходимо указать CHEN=1. В режиме самоинициализации при записи параметров в регистр CSR биты END и DONE недоступны.

Для запуска работы канала DMA в режиме с самоинициализацией необходимо в регистр CP записать адрес первого блока параметров DMA передачи. При этом 0 разряд записываемых данных должен содержать 1 (признак пуска самоинициализации). В результате этого, соответствующий канал загрузит в свои регистры параметры DMA передачи и начнет обмен данными.

После окончания передачи блока данных бит END в регистре CSR устанавливается в единичное состояние, если бит IM = 1 - выдается прерывание. По окончании передачи блока данных также проверяется состояние бита CHEN. Если он равен 1, то будет загружен следующий блок параметров DMA передачи и т.д. В противном случае цепочка DMA обменов закончится и в регистре CSR бит DONE установится в единичное состояние и выдается прерывание.

21.3.3.2 Программная инициализация

При необходимости каналы DMA могут инициализироваться программно. Для этого RISC должен загрузить все необходимые регистры индекса и смещения, а затем регистр CSR. При загрузке регистра CSR бит RUN необходимо установить в единичное состояние. Следует отметить, что бит RUN может быть использован для приостановки канала DMA. Для этого в любой момент времени в него необходимо записать 0. Для продолжения работы соответственно в бит RUN необходимо записать 1. Бит RUN может быть использован также для приостановки выполнения цепочки, если при загрузке очередных параметров он будет равен 0. Для продолжения выполнения цепочки в бит RUN необходимо записать 1. Для удобства организации обмена только с битом RUN выделен персональный адрес в адресном пространстве канала DMA.

21.4 Схема подключения

На Рисунок 21.78 представлена схема коммутации блоков DFE с контактными площадками, предназначенными для подключения АЦП/ЦАП (LVDS)

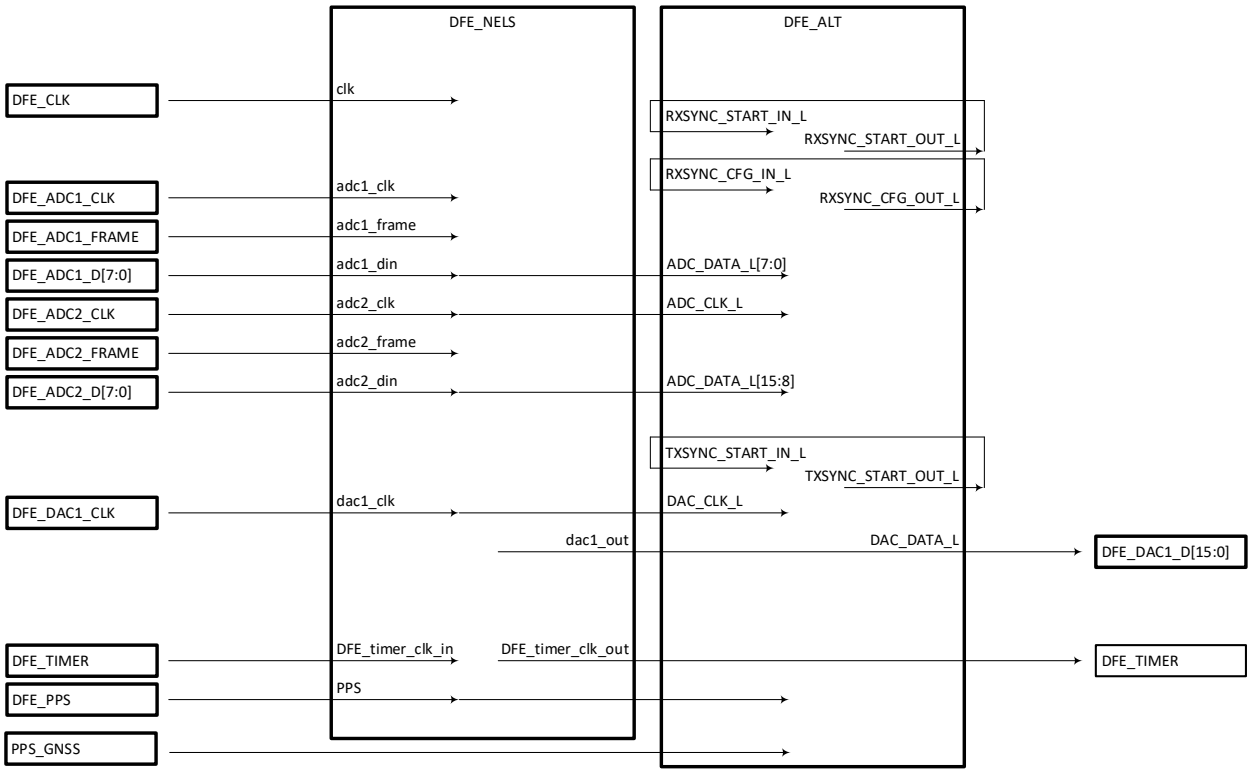


Рисунок 21.78. Схема коммутации блоков DFE с КП1

На Рисунок 21.79 представлена схема коммутации блоков DFE с контактными площадками, предназначенными для подключения АЦП/ЦАП (CMOS), SPI и GPIO

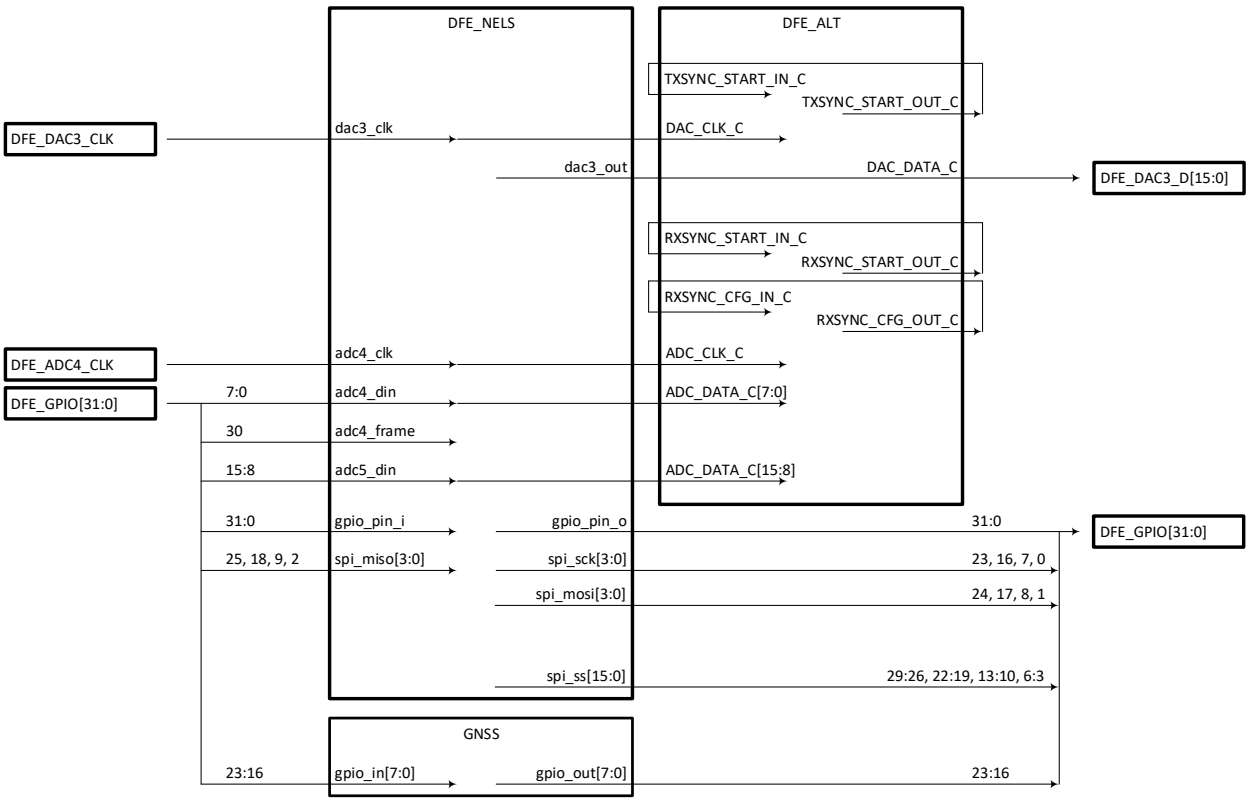


Рисунок 21.79. Схема коммутации блоков DFE с КП2

На Рисунок 21.80 представлена схема коммутации блоков DFE с контроллером Jesd204b, предназначенным для подключения АЦП/ЦАП (Jesd204b)

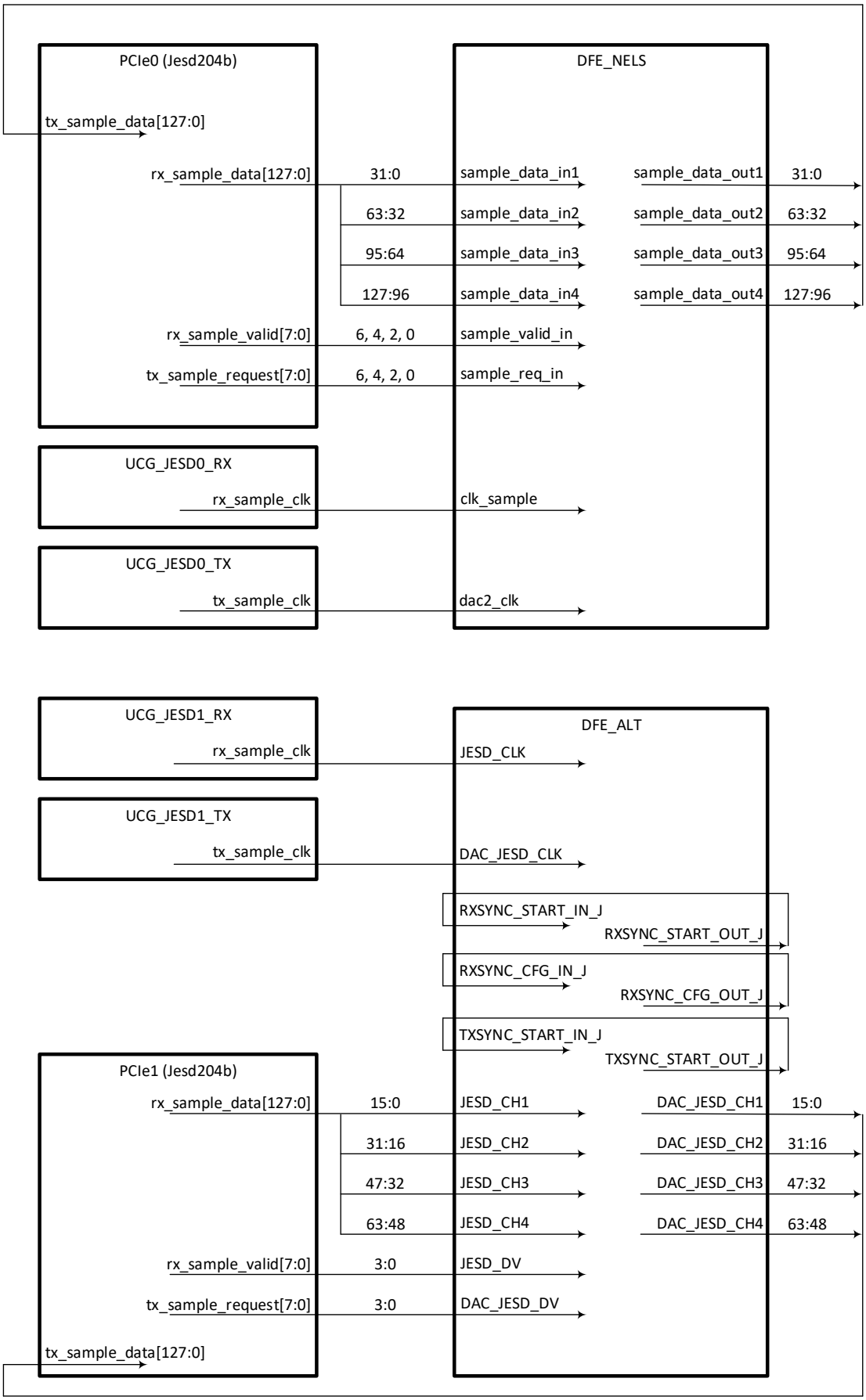


Рисунок 21.80. Схема коммутации блоков DFE с контроллером Jesd204b

22. УСКОРИТЕЛИ ЦИФРОВОЙ ОБРАБОТКИ СИГНАЛОВ

22.1 Общая информация

В состав микросхемы входит несколько аппаратных блоков, предназначенных для ускорения операций цифровой обработки сигналов в системах связи, в частности, при использовании микросхемы К1892ВМ21Я в базовых станциях:

- витерби декодер (VITDEC);
- сверточный кодер (CENC);
- турбо декодер (TDC);
- турбо кодер (TEC);
- блок быстрого преобразования Фурье (FFT).

По тактированию и подключению все ускорители условно разделены на три группы: блоки VITDEC, CENC – группа ACC0, блоки TDC, TEC – группа ACC1, блок FFT – группа ACC2. Каждая группа ускорителей имеет свой независимый домен для тактирования, а также интерфейс AXI для подключения к коммутатору подсистемы.

Тактовые домены для всех групп ускорителей перечислены в Таблица 22.1.

Таблица 22.1. Тактовые домены ускорителей

#	Домен	Описание
1	ACC0	Тактовый домен группы ускорителей ACC0
2	ACC1	Тактовый домен группы ускорителей ACC1
3	ACC2	Тактовый домен группы ускорителей ACC2
4	AXI	Общий тактовый домен для внешних интерфейсов AXI и сигналов событий (events)
5	CFG_AXI	Общий тактовый домен для внешнего конфигурационного интерфейса AXI и сигналов прерываний (irq)

На Рисунок 22.1, Рисунок 22.2 также изображены тактовые домены, блоки синхронизации и локальные коммутаторы интерфейсов AXI.

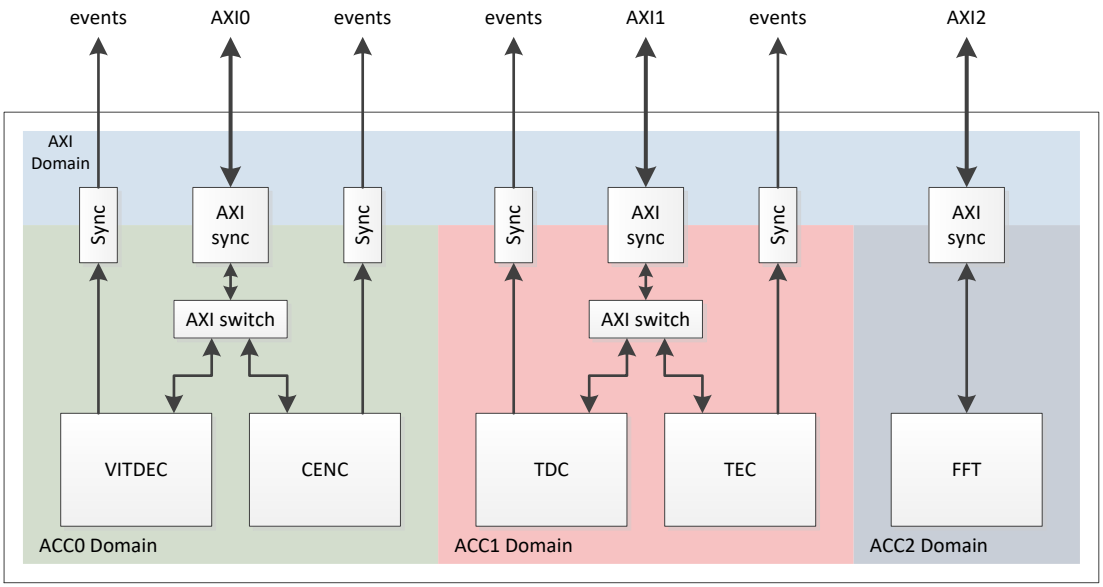


Рисунок 22.1. Тактовые домены ускорителей (интерфейсы данных)

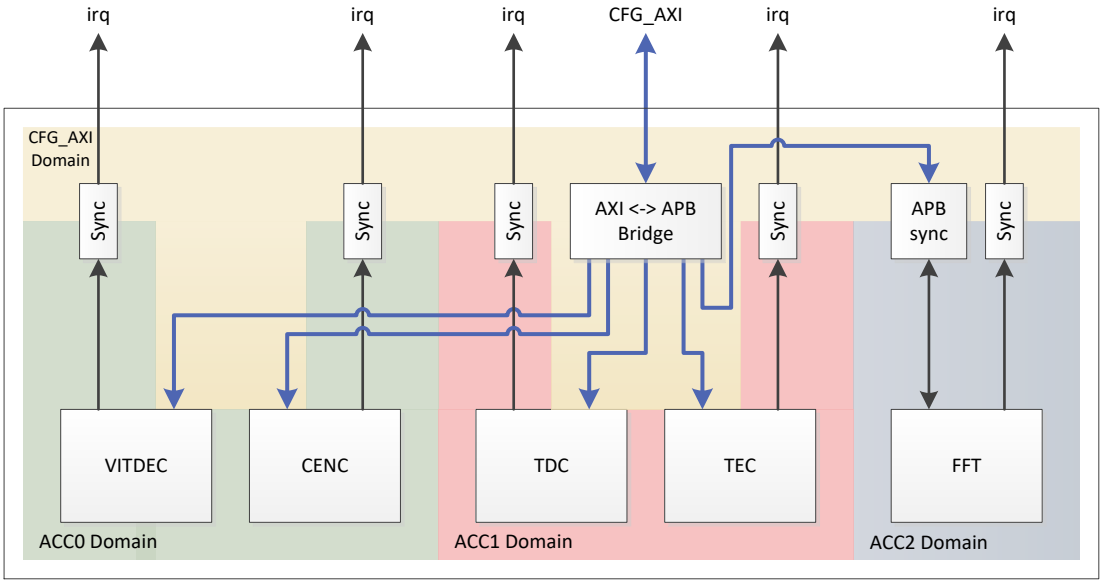


Рисунок 22.2. Тактовые домены ускорителей (конфигурационный интерфейс)

В следующих разделах дано подробное описание каждого ускорителя.

22.2 Витерби декодер (VITDEC)

22.2.1 Общая информация

22.2.1.1 Основные особенности

Блок VITDEC предназначен для декодирования сверточных несистематических нерекурсивных кодов по алгоритму Витерби.

Основные особенности VITDEC:

- Конфигурируемое количество порождающих полиномов: 2 – 8
- Конфигурируемое кодовое ограничение: 3 – 9
- Поддержка мягких входных оценок
- Поддержка перфорированных кодов
- Размер окна декодирования, бит: 192 – 1536
- Результат – жесткие решения в прямом порядке
- Поддерживаются только сверточные коды с терминацией в нулевое состояние
- Аппаратный интерфейс для работы с DMA
- Двухбуферная схема загрузки и выгрузки данных
- Производительность: не менее 50 Мбит/с @ 600 МГц

22.2.1.2 Описание алгоритма декодирования

На Рисунок 22.3, для примера, изображена схема сверточного кодера, а на Рисунок 22.4 – соответствующий граф переходов. Выход out[0] является функцией XOR от входа in, первого и второго разрядов сдвигового регистра. Выход out[1] является функцией XOR от входа и второго разряда сдвигового регистра. В каждый момент времени кодер пребывает в одном из состояний, определяемом значениями разрядов сдвигового регистра. Каждому переходу между состояниями соответствует некоторое значение сверточного кода, выдаваемое кодером. Декодирование Витерби происходит путем нахождения наиболее вероятного пути через возможные состояния кодера.

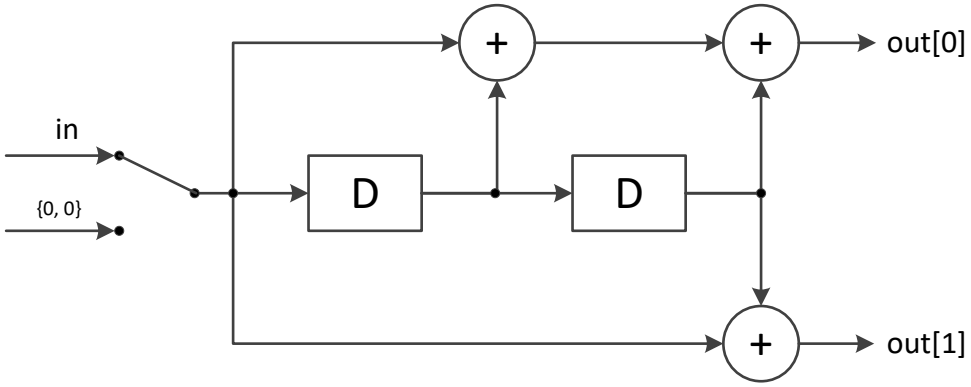


Рисунок 22.3. Пример схемы сверточного кодера

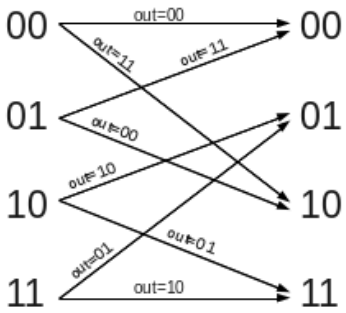


Рисунок 22.4. Граф переходов сверточного кодера

Основные параметры кодера:

- Длина кодового ограничения K, равна длине сдвигового регистра минус 1. На Рисунок 22.4 показан кодер с K = 3.
- Скорость кода R_0, определяемая как $R_0 = k/n$, где n – количество выходных бит, получающихся при кодировании k информационных бит. На Рисунок 22.4 показан кодер со скоростью ½.
- Генерирующие (порождающие) полиномы G(x) – описывают структуру кодера. Ненулевые разряды соответствуют стадиям сдвигового регистра, участвующим в формировании выходного сигнала. В представленном примере выход out[0] кодера имеет полином (111), выход out[1] имеет полином (101) или в восьмеричной системе $(G_0, G_1) = (7, 5)$.

Оценка стоимости переходов

Для каждого возможного перехода между состояниями декодер имеет эталонные значения, с которыми сравнивается входной сигнал. Стоимость переходов оценивается на основе расстояния Хэмминга между входным сигналом и эталоном. На Рисунок 22.5 отмечены стоимости переходов для входного сигнала '00' для рассматриваемого сверточного кода.

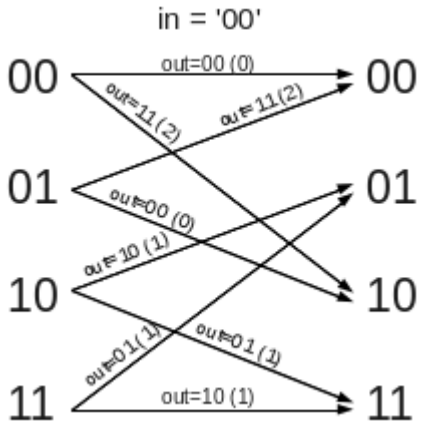


Рисунок 22.5. Стоимости переходов для всех состояний кодера

Операция ACS (Add Compare Select)

Для каждого нового состояния выбирается наиболее вероятный путь с учетом метрики предыдущего состояния и вычисленных метрик переходов. Метрики переходов скаладываются с метриками соответствующих предыдущих состояний. Результаты сравниваются, и делается выбор в пользу наименьшего. Данная операция получила название Add-Compare-Select (сложение-сравнение-выбор). На Рисунок 22.6 показана операция ACS. Сплошными линиями отмечены выбранные пути, пунктирными – отброшенные.

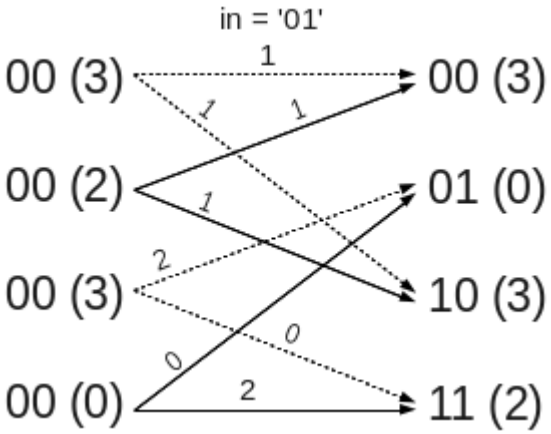


Рисунок 22.6. Пример операции ACS

В результате операции для каждого нового состояния формируется бит решения (decision bit) и новая метрика. Операция повторяется n раз. На каждом шаге метрики состояний перезаписываются, а биты решений сохраняются в специальную память, формируя так

называемую треллис-диаграмму или просто треллис. Параметр n называется глубиной треллиса.

На Рисунок 22.7 показан пример треллиса глубиной 7 для сверточного кода (7, 5). Узлы являются состояниями кодера, ребра – переходами между состояниями. На 5-м шаге поступают искаженные данные, однако накопленные метрики путей позволяют безошибочно продолжить составление треллиса.

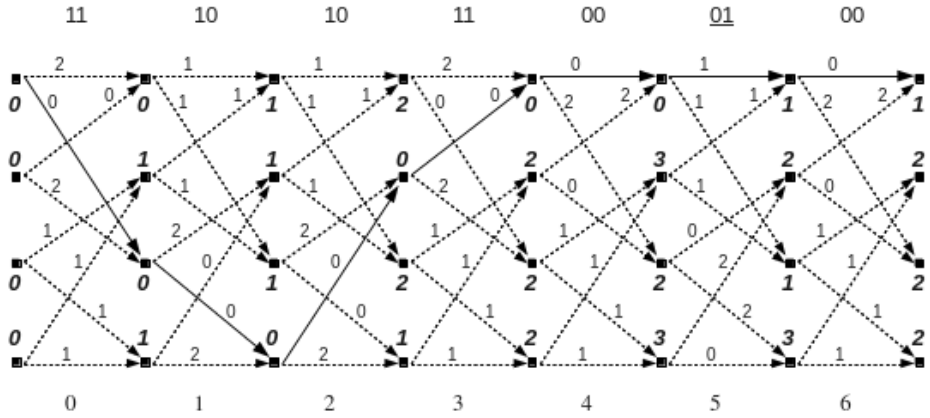


Рисунок 22.7. Треллис-диаграмма

Отслеживание выжившего пути (traceback)

После заполнения треллиса выбирается путь с наименьшей накопленной метрикой (выживший путь), необходимый для начала декодирования треллиса. Декодирование выполняется в обратном порядке. Зная конечное состояние выжившего пути и соответствующий бит решения, находится предыдущее состояние. Операция повторяется, пока не будет достигнуто начало треллиса. Зная последовательность состояний, в которых был кодер, можно найти последовательность, поданную на его вход, то есть, декодировать данные.

В приведенном выше примере треллис можно декодировать безошибочно, если начать отслеживание из последнего состояния (после шестого шага треллиса).

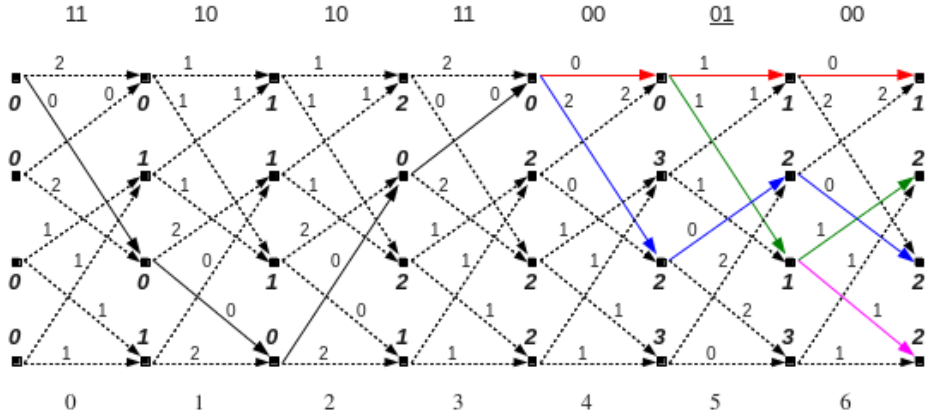


Рисунок 22.8. Треллис-диаграмма с отслеживанием нескольких путей

Если начинать отслеживание после пятого шага, то это, скорее всего, приведет к ошибке, поскольку невозможно однозначно выбрать выживший путь (пути 0 и 2 обладают одинаковыми метриками). Чтобы избежать ошибочного декодирования в подобных случаях, используется т. н. тренировочная последовательность. Треллис заполняется еще на несколько шагов, затем эта дополнительная часть используется для определения выжившего состояния. На Рисунок 22.8 показано отслеживание пути для нескольких конечных состояний. Как видно, вне зависимости от выбранного начального состояния, отслеживаемый путь сходится с остальными в некоторой точке. Можно использовать последние 3 шага треллиса для определения выжившего пути, в этом случае данная часть треллиса будет тренировочной последовательностью для декодирования оставшейся части. Использование тренировочной последовательности позволяет улучшить значение BER (Bit Error Rate) декодированных данных, но, как правило, ценой снижения производительности декодера. Данный подход возможен, если на один пакет декодируемых данных приходится больше одного треллиса. Тогда начало следующего треллиса используется в качестве тренировочной последовательности для текущего.

Тренировочная последовательность не может быть использована в самом конце треллиса, т.к. больше нет данных для дополнения решетки. В этом случае может использоваться т.н. терминация кодера – приведение кодера в известное состояние после кодирования входного сообщения.

VITDEC поддерживает только коды с терминацией в нулевое состояние. Для этого после кодирования сообщения на вход кодера подается последовательность из (K - 1) нулей. При этом на выходе кодера будет получено $(K - 1)/R_0$ «хвостовых» бит.

Таким образом, если F – размер исходного сообщения (в битах), то полное количество бит в закодированном блоке равно $N_{ENC} = (F + (K – 1))/R_0$. Параметр N_{ENC} задается с помощью поля ENC_BLK_LEN регистра **VITDEC_BLKCFG4**.

22.2.2 Структурная схема

На Рисунок 22.9 изображена структурная схема VITDEC.

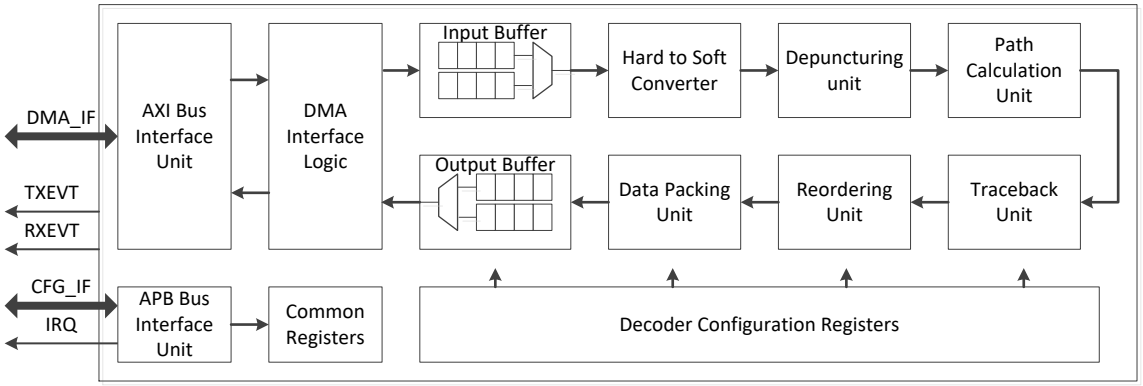


Рисунок 22.9. Структурная схема VITDEC

Основные составляющие блоки VITDEC и их функции:

- AXI Bus Interface Unit: блок реализует slave-устройство на интерфейсе данных DMA_IF.
- APB Bus Interface Unit: блок реализует slave-устройство на конфигурационном интерфейсе CFG_IF.
- DMA Interface Logic: реализует управление загрузкой, выгрузкой данных и конфигурации по интерфейсу DMA_IF.
- Input Buffers, Output Buffers: буферы FIFO для хранения входных и выходных данных соответственно. Для ускорения работы декодера в каждом направлении предусмотрено два буфера. Переключение буферов выполняется автоматически. Размер каждого буфера – 16 слов по 64 бита.
- Hard to Soft Converter: блок преобразует жесткие входные оценки во внутренний формат. Мягкие входные оценки пропускаются без изменений.
- Depuncturing Unit: осуществляет выборку входных данных в соответствии с используемым шаблоном перфорации.
- Path Calculation Unit: вычисляет метрики путей, использующиеся при составлении треллис-диаграммы. Блок может обрабатывать до 32 метрик за 1 такт. Для кодового ограничения 3, 4, 5, 6 это означает генерацию одного шага треллиса за такт. Для кодовых ограничений 7, 8, 9 один шаг треллиса генерируется за 2, 4 и 8 тактов соответственно.
- Traceback Unit: содержит память, в которую записывается треллис-диаграмма, а также устройство отслеживания выжившего пути.
- Reordering Unit: восстанавливает прямой порядок следования решений, выдаваемых блоком Traceback Unit.
- Data Packing Unit: выполняет упаковку жестких решений в 64-х разрядные слова.
- Common Registers: общие регистры статуса, программного сброса и прерываний.
- Decoder Configuration Registers: регистры параметров декодирования.

22.2.3 Функционирование

22.2.3.1 Введение

Типовой сценарий использования блока VITDEC предполагает работу совместно с системным контроллером DMA посредством сигналов событий TXEVT и RXEVT, которые соответствуют двум направлениям передачи данных: TX – из системной памяти во внутреннюю память VITDEC (параметры декодирования и входные данные), RX – из памяти VITDEC в системную память (выходных данные).

Общий алгоритм использования блока VITDEC состоит из следующих шагов:

1. Хост-процессор (CPU) настраивает каналы контроллера DMA для передачи в VITDEC параметров и закодированного сообщения, а также для чтения декодированных данных из VITDEC.
2. После подготовки входных данных для декодирования CPU запускает контроллер DMA для передачи конфигурационных параметров декодируемого блока.
3. С помощью события TXEVT блок VITDEC сигнализирует контроллеру DMA о готовности к приему декодируемого блока данных. Прием осуществляется порциями по шестнадцать 64-х битных слов. Событие TXEVT генерируется для получения каждой порции данных.
4. VITDEC выполняет декодирование по мере поступления входных данных. Выходные данные помещаются в свободный буфер FIFO глубиной 16 слов по 64 бит. При заполнении буфера либо при завершении декодирования всего блока генерируется событие RXEVT, что запускает контроллер DMA на чтение декодированных данных из VITDEC.
5. После чтения всего декодированного блока контроллер DMA генерирует прерывание для CPU.
6. Если имеется очередной блок для декодирования, то выполняется переход к шагу 3.

Далее приводится более подробное описание этих шагов.

22.2.3.2 Работа с DMA

Настройка контроллера DMA заключается в подготовке дескрипторов задач для пересылки необходимой информации.

В направлении TX выполняются следующие пересылки:

- конфигурационные параметры (регистры **VITDEC_BLKCFG***). Необходимо выполнять запись всех регистров, даже если их значения не изменялись;
- блок входных данных (закодированное сообщение).

В направлении RX выполняется пересылка блока выходных данных (результат декодирования).

Блоки входных/выходных данных записываются/считываются частями (порциями). Размер части определяется размером входного/выходного буфера FIFO и составляет 16 слов по 64 бита (1024 бит). Размер последней части блока может быть меньше.

Для уведомления контроллера DMA о возможности выполнить пересылку одной части блока данных VITDEC использует сигналы TXEVT (направление TX) и RXEVT (направление RX), подключенные к каналам DMA следующим образом (см. Раздел 6.4.5 «Распределение событий»):

- TXEVT – DMA0 канал 7;
- RXEVT – DMA0 канал 6.

На Рисунок 22.10 показан пример выполняемых пересылок для декодирования одного блока. Здесь CFG – конфигурационные параметры, DI #i – части блока входных данных, DO #i – части блока выходных данных.

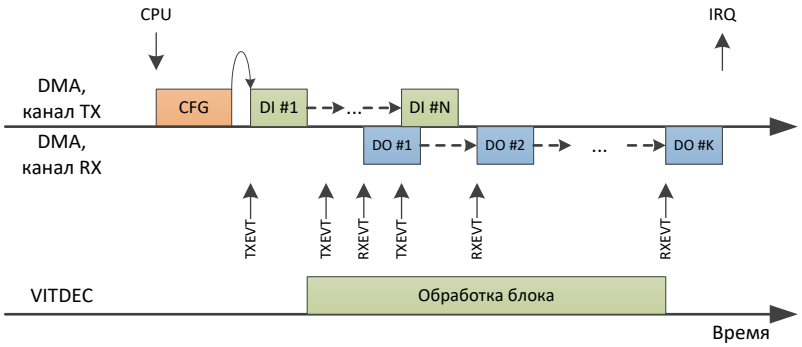


Рисунок 22.10. Пример декодирования одного блока

Для ускорения передачи данных используется возможность DMA объединять несколько пересылок в цепочки. Так, в примере выше передача конфигурационных параметров связана в цепочку с пересылкой входных данных.

В Таблица 22.2 приведены оптимальные значения параметров дескрипторов DMA для всех рассмотренных пересылок. В таблице используются следующие обозначения:

- N_{IN} – размер кодированного блока в байтах
- N_{OUT} – размер декодированного блока в байтах
- ceil(x) – операция округления x до большего целого

Таблица 22.2. Настройки контроллера DMA для VITDEC

Параметр DMA	Значение для параметров декодирования	Значение для входных данных	Значение для выходных данных
CSR.RUN	1	1	1
CSR.DIR	0	0	1

Параметр DMA	Значение для параметров декодирования	Значение для входных данных	Значение для выходных данных
CSR.REV	0	0	0
CSR.2D	0	0	0
CSR.WN	7	7	7
CSR.SIZE	4	4	4
CSR.IM	0	0	0
CSR.CHEN	1	0	0
CSR.MASK	0	1	1
IR0	Адрес конфигурационных параметров в системной памяти	Адрес блока входных данных в системной памяти	Адрес блока выходных данных в системной памяти
IR1	Адрес регистра VITDEC_BLKCFG0	Адрес VITDEC_IBUF	Адрес VITDEC_OBUF
OR0	1	1	1
OR1	1	0	0
ORY	0 (не используется)	0 (не используется)	0 (не используется)
WCX	1	ceil[N_{IN} / 16] - 1	ceil[N_{OUT} / 16] - 1
WCY	0 (не используется)	0 (не используется)	0 (не используется)
CP	Адрес дескриптора блока входных данных	0	0

Примечание: Контроллер DMA оперирует 128-ми разрядными словами, а интерфейс VITDEC имеет разрядность 64 бита. Это накладывает дополнительные ограничения на конфигурацию DMA – если количество 64-х разрядных слов записываемых во входной буфер VITDEC или считываемых из выходного буфера нечетное, то контроллер DMA должен пересылать последнее слово с размером 64 бита вместо 128-ми (CSR.SIZE = 3). При этом последнее передаваемое слово должно содержать маску NDMAR_OFF, в противном случае контроллер DMA будет ожидать сигнал, подтверждающий получение данных (RXEVT), и последнее слово не будет отправлено.

Новое декодирование может быть запущено после считывания данных предыдущего декодирования из VITDEC_OBUF.

22.2.3.3 Генерирующие полиномы

Декодер поддерживает сверточные коды, сформированные с использованием от двух до восьми генерирующих полиномов (POLY0 – POLY7). Каждый полином в общем виде определяется формулой $G(x) = b_8x^{-8}+b_7x^{-7}+b_6x^{-6}+b_5x^{-5}+b_4x^{-4}+b_3x^{-3}+b_2x^{-2}+b_1x^{-1}+b_0$. Бит b_1 соответствует наиболее значимому биту (MSB) POLYn, бит b_8 – наименее значимому биту (LSB) POLYn. Значения полиномов должны записываться в регистры, начиная с POLY0. Неиспользуемые поля POLYn должны быть заполнены значением 0x00.

Кодовое ограничение вычисляется автоматически по ненулевым значениям POLYn.

Например, для кода с кодовым ограничением K = 7 и полиномами (в восьмеричной системе) G_0 = 133, G_1 = 154, G_2 = 176 значения POLYn будут следующими: POLY0 = 0x6C, POLY1 = 0xB0, POLY2 = 0xF8, POLY3–7 = 0x00.

22.2.3.4 Перфорация

Для повышения скорости кода может быть использована перфорация, при этом часть закодированных бит удаляется и не передается в канал.

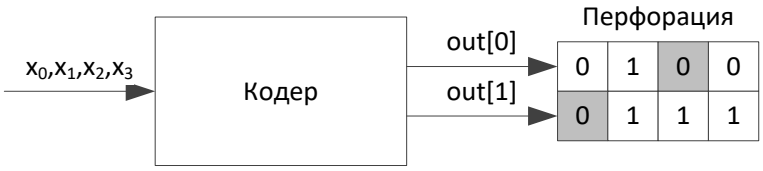


Рисунок 22.11. Перфорация сверточного кода

На Рисунок 22.11 приведен пример, перфорации сверточного кода с базовой скоростью 1/2. На вход кодера поступает сообщение размером 4 бита, на выходе кодера получается блок размером 8 бит (00110101). В результате перфорации исключаются 2 бита, т.е. новое значение скорости – 4/6 или 2/3, блок после перфорации – 011101.

В декодере реализована аппаратная поддержка перфорированных кодов. Кодированные данные поступают из входного буфера в блок Depuncturing Unit, в который предварительно записан шаблон перфорации. Выбранные в соответствии с шаблоном данные используются для вычисления метрик переходов, удаленные биты не вносят вклад в вычисление.

Шаблон перфорации задается полями PPATERN0 – PPATERN7 (регистры **VITDEC_BLKCFG2**, **VITDEC_BLKCFG3**). Каждый бит PPATERNn[i] соответствует выходу кодера out[i], при этом PPATERNn[i] = 0 означает, что бит i был удален. Количество ненулевых полей PPATERNn, начиная с PPATERN0, определяет размер цикла перфорации.

Для примера на Рисунок 22.11 значения PPATERNn будут следующими: PPATERN0 = 0x01, PPATERN1 = 0x03, PPATERN2 = 0x02, PPATERN3 = 0x03, PPATERN4–7 = 0x00.

Если перфорация не используется, то PPATERN0 устанавливается в соответствии со скоростью кода (0x03 для 1/2, 0x07 для 1/3, …, 0xFF для 1/8), поля PPATERN1–7 записываются нулями.

При использовании перфорации количество бит во входном блоке также определяется по формуле $N_{ENC} = (F + (K – 1))/R$. Здесь R – скорость кода с учетом перфорации (например, R = 2/3).

22.2.3.5 Формат входных метрик

Декодер позволяет работать как с жесткими, так и с мягкими оценками входных данных. Формат задается полем IFMT регистра **VITDEC_BLKCFG5**.

Жесткие оценки имеют разрядность один бит и записываются во входной буфер VITDEC упакованными в 64-х разрядные слова. При этом первый бит закодированного блока Y(0) соответствует наименее значимому разряду слова. Если размер закодированного сообщения не кратен 64, то старшие разряды последнего слова не используются (Таблица 22.3).

Таблица 22.3. Формат жестких оценок

Бит					Номер 64-х разрядного слова
63	…	2	1	0	
-	-	$Y(N_{ENC}-1)$	$Y(N_{ENC}-2)$	…	$ceil(N_{ENC}/64)-1$
…	…	…	…	…	…
Y(127)	…	…	Y(65)	Y(64)	1
Y(63)	…	…	Y(1)	Y(0)	0

При работе с мягкими оценками (метриками) каждая метрика задается 8-ми разрядным знаковым числом. Формат упаковки мягких оценок в слова показан в Таблица 22.4.

Таблица 22.4. Формат мягких оценок

Бит					Номер 64-х разрядного слова
63:56	…	…	15:8	7:0	
-	-	$Y(N_{ENC} - 1)$	$Y(N_{ENC} - 2)$	…	$ceil(N_{ENC}*8/64)-1$
…	…	…	…	…	…
Y(15)	…	…	Y(65)	Y(64)	1
Y(7)	…	…	Y(1)	Y(0)	0

Допустимый диапазон значений метрик зависит от базовой скорости используемого кода R_0 (т.е. скорости кода до перфорации). Возможные варианты приведены в Таблица 22.5. Максимальное положительное значение соответствует наиболее вероятной единице, наименьшее отрицательное – наиболее вероятному нулю.

Таблица 22.5. Диапазоны значений мягких оценок

Базовая кодовая скорость R_0	Диапазон значений
1/8	[-31; 31]
1/7	[-36; 36]
1/6	[-42; 42]
1/5	[-51; 51]
1/4	[-63; 63]
1/3	[-85; 85]
1/2	[-127; 127]

22.2.3.6 Глубина декодирования

Память блока Traceback Unit может хранить до 49152 бит жестких решений, при этом на каждый шаг треллиса приходится $2^{(K-1)}$ бит. Таким образом, в память могут быть записаны решения по $49152/2^{(K-1)}$ шагам треллиса. Если F – размер исходного сообщения, а K – длина кодового ограничения, то значения K и F определяют, поместятся ли решения для всех символов сообщения в памяти или нет.

Если биты решений для всего треллиса могут быть сохранены в памяти Traceback Unit, то поиск выжившего пути выполняется один раз для всего сообщения (Рисунок 22.12). В этом случае параметр WIN_MODE = 0.



Рисунок 22.12. Декодирование при WIN_MODE=0

В противном случае декодирование блока производится перекрывающимися окнами размером WL = D + C, а параметр WIN_MODE = 1 (Рисунок 22.13).

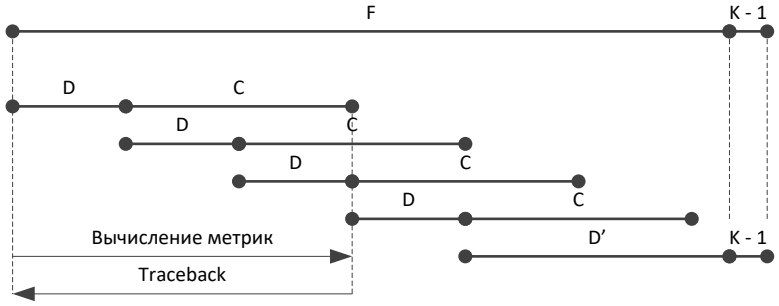


Рисунок 22.13. Декодирование при WIN_MODE=1

Значение параметра WIN_MODE задается в регистре **VITDEC_BLKCFG5** и определяется следующим образом:

$$WIN\_MODE = \begin{cases} 0, & F + (K-1) \le WLmax \\ 1, & F + (K-1) > WLmax \end{cases}$$

Здесь F – размер исходного сообщения в битах, WLmax – максимальная глубина декодирования для заданного кодового ограничения K (Таблица 22.6).

Таблица 22.6. Максимальная глубина декодирования

K	WLmax
9	192
8	384
7	768
6	1536
5	1536
4	1536
3	1536

При WIN_MODE = 1 во время выполнения операции traceback на выход декодера передается только D декодированных бит (Рисунок 22.13). Часть окна размером C бит используется как тренировочная последовательность. Размер последнего окна может быть меньше остальных.

В декодере VITDEC значения D и C, а значит и размер окна, определяются параметром DEC_LEN:

- D = DEC_LEN + 1;
- C = 2*D;
- WL = 3*D = 3*(DEC_LEN + 1).

Значение C необходимо выбирать достаточно большим, как правило, не менее 5K. При этом должно выполняться условие WL <= WLmax.

При WIN_MODE = 0 декодирование выполняется без использования окон, и значение DEC_LEN не используется.

22.2.3.7 Программный сброс

При необходимости, программный сброс выполняется путем записи значения 1 в регистр **VITDEC_SWRST**. Программный сброс приводит VITDEC в начальное состояние и не влияет на состояние регистров **VITDEC_BLKCFG***. Во время выполнения сброса все каналы DMA, работающие с VITDEC, должны быть остановлены, а также не должно быть активных транзакций на интерфейсе DMA_IF.

22.2.3.8 Прерывания

Аппаратные сигналы TXEVT, RXEVT также могут генерировать прерывание. Эту возможность можно использовать для отладки или для альтернативного (более медленного) способа работы с VITDEC без участия контроллера DMA.

Для использования прерываний необходимо установить соответствующие разряды в регистре **VITDEC_IRQ_EN**. Состояние разрешенных прерываний отображается в регистре **VITDEC_IRQ_EN_STAT**. Для сброса активного прерывания необходимо записать '1' в соответствующий разряд регистра **VITDEC_IRQ_CLR**.

22.2.4 Регистры

22.2.4.1 Перечень регистров

В Таблица 22.7 перечислены общие регистры статуса и управления VITDEC. Доступ к регистрам возможен через интерфейс CFG_IF только 32-х разрядными словами.

Таблица 22.7. Регистры статуса и управления VITDEC (интерфейс CFG_IF)

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
VITDEC_REVID	Версия RTL	0x18[1]	RO	0x0000
VITDEC_SWRST	Регистр программного сброса	0x00	WO	0x0004
VITDEC_STAT	Регистр статуса декодера	0x31	RO	0x000C
VITDEC_TXES	Регистр установки события TXEVT	0x00	WO	0x0010
VITDEC_RXES	Регистр установки события RXEVT	0x00	WO	0x0014
VITDEC_IRQ_EN	Регистр разрешения прерываний	0x0	RW	0x0018
VITDEC_IRQ_STAT	Регистр статуса/установки прерываний	0x0	RW	0x001C
VITDEC_IRQ_EN_STAT	Регистр статуса разрешенных прерываний	0x0	RO	0x0020
VITDEC_IRQ_CLR	Регистр сброса прерываний	0x0	WO	0x0024

Примечания:
1) Версия RTL имеет формат 0xXY и соответствует версии SVN релиза. Например, для релиза r1p2 X = 1, Y = 2, значение регистра = 0x12

В Таблица 22.8 перечислены регистры для настройки параметров декодирования. Доступ к регистрам производится по интерфейсу DMA_IF. Поддерживаются 32-х, 64-х и 128-ми разрядные обращения. Запись по неиспользуемым адресам игнорируется.

Таблица 22.8. Регистры настройки декодера VITDEC (интерфейс DMA_IF)

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
VITDEC_BLKCFG0	Конфигурационный регистр 0	0x0	RW	0x0000
VITDEC_BLKCFG1	Конфигурационный регистр 1	0x0	RW	0x0004
VITDEC_BLKCFG2	Конфигурационный регистр 2	0x0	RW	0x0008
VITDEC_BLKCFG3	Конфигурационный регистр 3	0x0	RW	0x000C
VITDEC_BLKCFG4	Конфигурационный регистр 4	0x0	RW	0x0010
VITDEC_BLKCFG5	Конфигурационный регистр 5	0x0	RW	0x0014
VITDEC_IBUF	Регистр записи буфера FIFO входных данных	0x0	WO	0x0080
VITDEC_OBUF	Регистр чтения буфера FIFO выходных данных	0x0	RO	0x00C0

22.2.4.2 Регистры статуса и управления

22.2.4.2.1 Регистр VITDEC_REVID

Таблица 22.9. Формат регистра VITDEC_REVID

Номер бита	Условное обозначение	Назначение
31:8	-	Резерв
7:4	X	Старшая часть номера версии (X)

Номер бита	Условное обозначение	Назначение
3:0	Y	Младшая часть номера версии (Y)

22.2.4.2.2 Регистр VITDEC_SWRST

Таблица 22.10. Формат регистра VITDEC_SWRST

Номер бита	Условное обозначение	Назначение
31:1	-	Резерв
0	RESET	Запись 1 в этот бит инициирует программный сброс блока. Сброс не влияет на состояние регистров **VITDEC_BLKCFG**

22.2.4.2.3 Регистр VITDEC_STAT

Таблица 22.11. Формат регистра VITDEC_STAT

Номер бита	Условное обозначение	Назначение
31:8	-	Резерв
7:6	OBUF_BUSY	Поле OBUF_BUSY[1:0] содержит статус наполненности выходных буферов FIFO. OBUF_BUSY[i]: 0: буфер i не полон; 1: буфер i полон
5:4	IBUF_EMPTY	Поле IBUF_EMPTY[1:0] содержит статус наполненности входных буферов FIFO. IBUF_EMPTY[i]: 0: буфер i не пуст; 1: буфер i пуст
3:2	-	Резерв
1	WAIT_TX_DATA	Значение 1 этого бита означает, что декодер находится в состоянии ожидания загрузки входных данных
0	WAIT_BLKCFG	Значение 1 этого бита означает, что декодер находится в состоянии ожидания записи регистров конфигурации **VITDEC_BLKCFG**

22.2.4.2.4 Регистр VITDEC_TXES

Таблица 22.12. Формат регистра VITDEC_TXES

Номер бита	Условное обозначение	Назначение
31:1	-	Резерв
0	TXEVT	Запись 1 в этот бит генерирует событие TXEVT. Используется для отладки

22.2.4.2.5 Регистр VITDEC_RXES

Таблица 22.13. Формат регистра VITDEC_RXES

Номер бита	Условное обозначение	Назначение
31:1	-	Резерв
0	RXEVT	Запись 1 в этот бит генерирует событие RXEVT. Используется для отладки

22.2.4.2.6 Регистр VITDEC_IRQ_EN

Таблица 22.14. Формат регистра VITDEC_IRQ_EN

Номер бита	Условное обозначение	Назначение
31:2	-	Резерв

Номер бита	Условное обозначение	Назначение
1	RXEVT	Разрешение прерывания по событию RXEVT
0	TXEVT	Разрешение прерывания по событию TXEVT

22.2.4.2.7 Регистр VITDEC_IRQ_STAT

Регистр содержит биты статуса для всех прерываний. Бит статуса устанавливается в 1 при возникновении прерывания без учета разрешений, либо при записи 1 в этот бит (эта возможность может быть использована для отладочных целей). Запись 0 игнорируется.

Таблица 22.15. Формат регистра VITDEC_IRQ_STAT

Номер бита	Условное обозначение	Назначение
31:2	-	Резерв
1	RXEVT	Статус прерывания по событию RXEVT
0	TXEVT	Статус прерывания по событию TXEVT

22.2.4.2.8 Регистр VITDEC_IRQ_EN_STAT

Регистр содержит биты статуса для всех разрешенных прерываний. Бит статуса устанавливается в 1 при возникновении прерывания только, если установлен соответствующий бит в регистре **VITDEC_IRQ_EN**. Регистр доступен только для чтения.

Таблица 22.16. Формат регистра VITDEC_IRQ_EN_STAT

Номер бита	Условное обозначение	Назначение
31:2	-	Резерв
1	RXEVT	Статус прерывания по событию RXEVT
0	TXEVT	Статус прерывания по событию TXEVT

22.2.4.2.9 Регистр VITDEC_IRQ_CLR

Регистр предназначен для программного сброса прерываний, бит статуса которых установлен в регистре **VITDEC_IRQ_STAT**. Сброс производится путем записи 1 в соответствующий бит. Запись 0 игнорируется. При чтении регистра возвращается 0.

Таблица 22.17. Формат регистра VITDEC_IRQ_CLR

Номер бита	Условное обозначение	Назначение
31:2	-	Резерв
1	RXEVT	Сброс прерывания по событию RXEVT
0	TXEVT	Сброс прерывания по событию TXEVT

22.2.4.3 Регистры настройки декодера

22.2.4.3.1 Регистр VITDEC_BLKCFG0

Таблица 22.18. Формат регистра VITDEC_BLKCFG0

Номер бита	Условное обозначение	Назначение
31:24	POLY3	Генерирующий полином G_3
23:16	POLY2	Генерирующий полином G_2
15:8	POLY1	Генерирующий полином G_1
7:0	POLY0	Генерирующий полином G_0

22.2.4.3.2 Регистр VITDEC_BLKCFG1

Таблица 22.19. Формат регистра VITDEC_BLKCFG1

Номер бита	Условное обозначение	Назначение
31:24	POLY7	Генерирующий полином G_7
23:16	POLY6	Генерирующий полином G_6
15:8	POLY5	Генерирующий полином G_5
7:0	POLY4	Генерирующий полином G_4

22.2.4.3.3 Регистр VITDEC_BLKCFG2

Таблица 22.20. Формат регистра VITDEC_BLKCFG2

Номер бита	Условное обозначение	Назначение
31:24	PPATERN3	Шаблон перфорации, строка 3
23:16	PPATERN2	Шаблон перфорации, строка 2
15:8	PPATERN1	Шаблон перфорации, строка 1
7:0	PPATERN0	Шаблон перфорации, строка 0

22.2.4.3.4 Регистр VITDEC_BLKCFG3

Таблица 22.21. Формат регистра VITDEC_BLKCFG3

Номер бита	Условное обозначение	Назначение
31:24	PPATERN7	Шаблон перфорации, строка 7
23:16	PPATERN6	Шаблон перфорации, строка 6
15:8	PPATERN5	Шаблон перфорации, строка 5
7:0	PPATERN4	Шаблон перфорации, строка 4

22.2.4.3.5 Регистр VITDEC_BLKCFG4

Таблица 22.22. Формат регистра VITDEC_BLKCFG4

Номер бита	Условное обозначение	Назначение
31:20	-	Резерв
19:0	ENC_BLK_LEN	Задает значение размера закодированного блока N_{ENC} в битах (с учетом «хвостовых» бит). ENC_BLK_LEN = N_{ENC} - 1 = (F + K - 1)/R - 1

22.2.4.3.6 Регистр VITDEC_BLKCFG5

Таблица 22.23. Формат регистра VITDEC_BLKCFG5

Номер бита	Условное обозначение	Назначение
31:25	-	Резерв
24:16	DEC_LEN	Значение размера декодируемой части окна минус 1 (D - 1). Используется только при WIN_MODE = 1.
15:2	-	Резерв
1	WIN_MODE	Включение режима декодирования: 0: оконный режим выключен; 1: оконный режим включен
0	IFMT	Формат входных оценок: 0: жесткие оценки; 1: мягкие оценки

22.3 Сверточный кодер (CENC)

22.3.1 Общая информация

22.3.1.1 Основные особенности

Блок CENC предназначен для выполнения помехоустойчивого кодирования данных с использованием сверточного кода.

Основные особенности CENC:

- конфигурируемое количество порождающих полиномов: 2 ÷ 8;
- конфигурируемое кодовое ограничение: 3 ÷ 9;
- поддержка перфорированных кодов;
- аппаратный интерфейс для работы с dma;
- двухбуферная схема загрузки и выгрузки данных;
- производительность: не менее 50 мбит/с @ 600 мгц;
- поддерживаются только сверточные коды с терминацией в нулевое состояние.

22.3.1.2 Структура кода

В блоке CENC реализован сверточный кодер, показанный на Рисунок 22.14.

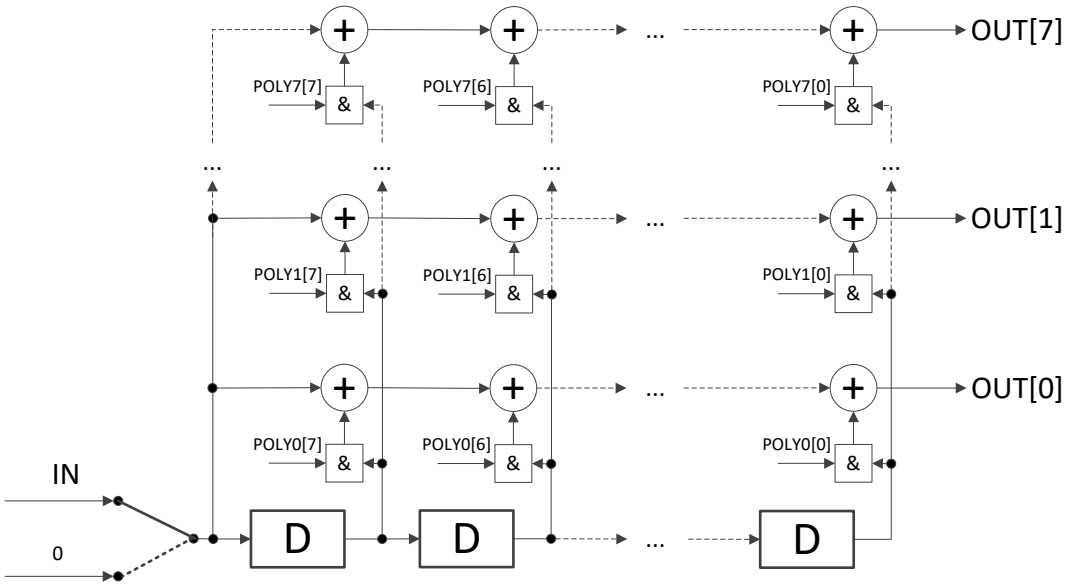


Рисунок 22.14. Структура кодера CENC

На вход **IN** кодера поступает блок данных (исходное сообщение) размером N_{IN} бит. На каждый входной бит на выходе кодера **OUT** формируется символ. Количество бит в символе определяется базовой скоростью кода и задается ненулевыми полиномами POLYn. Длина кодера определяется кодовым ограничением K.

После кодирования входного сообщения кодер переключается в режим терминации для перехода в нулевое состояние. При этом на выходе формируется (K – 1) «хвостовых» символа.

22.3.2 Структурная схема

На Рисунок 22.15 изображена структурная схема CENC.

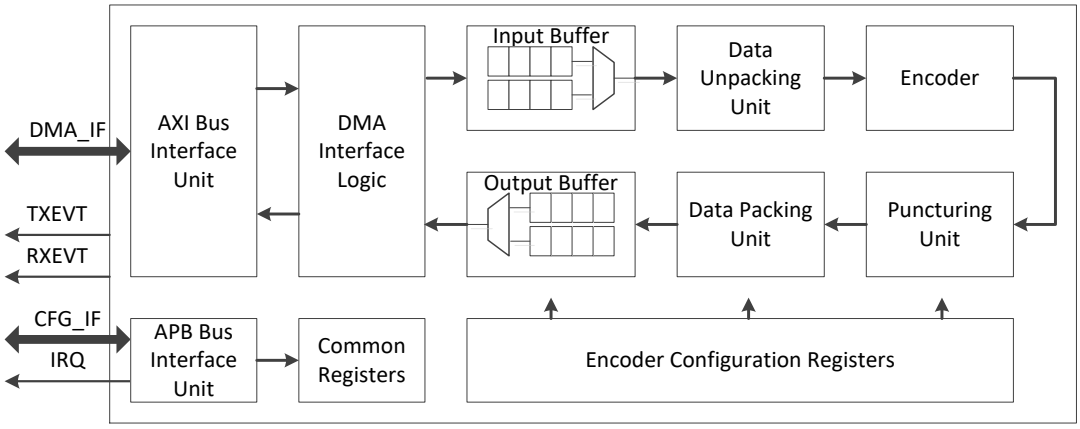


Рисунок 22.15. Структурная схема CENC

Основные составляющие блоки CENC и их функции:

- AXI Bus Interface Unit: блок реализует slave-устройство на интерфейсе данных DMA_IF.
- APB Bus Interface Unit: блок реализует slave-устройство на конфигурационном интерфейсе CFG_IF.
- DMA Interface Logic: реализует управление загрузкой, выгрузкой данных и конфигурации по интерфейсу DMA_IF.
- Input Buffers, Output Buffers: буферы FIFO для хранения входных и выходных данных соответственно. Для ускорения работы кодера в каждом направлении предусмотрено два буфера. Переключение буферов выполняется автоматически. Размер каждого буфера – 16 слов по 64 бита.
- Data Unpacking Unit: блок выполняет распаковку 64-разрядных слов в последовательность бит.
- Encoder: выполняет сверточное кодирование.
- Punturing Unit: блок выполняет перфорацию закодированных данных.
- Data Packing Unit: выполняет упаковку закодированных данных в 64-х разрядные слова.
- Common Registers: общие регистры статуса, программного сброса и прерываний.
- Encoder Configuration Registers: регистры параметров кодирования.

22.3.3 Функционирование

22.3.3.1 Введение

Типовой сценарий использования блока CENC предполагает работу совместно с системным контроллером DMA посредством сигналов событий TXEVT и RXEVT, которые соответствуют двум направлениям передачи данных: TX – из системной памяти во внутреннюю память CENC (параметры кодирования и входные данные), RX – из памяти CENC в системную память (выходных закодированные данные).

Общий алгоритм использования блока CENC состоит из следующих шагов:

1. Хост-процессор (CPU) настраивает системный блок DMA для передачи в CENC параметров и данных кодируемого блока (или набора из BLK_CNT + 1 блоков), а также для чтения кодированных блоков из CENC.
2. После подготовки входных данных для кодирования CPU запускает контроллер DMA для передачи конфигурационных параметров кодируемого блока (или набора из BLK_CNT + 1 блоков).
3. С помощью события TXEVT кодер CENC сигнализирует контроллеру DMA о готовности к приему блока данных для кодирования. Прием осуществляется порциями по шестнадцать 64-х битных слов. Событие TXEVT генерируется для получения каждой порции данных.
4. CENC выполняет кодирование по мере поступления входных данных. Выходные данные помещаются в свободный буфер FIFO глубиной 16 слов по 64 бит. При заполнении буфера либо при завершении кодирования всего блока генерируется событие RXEVT, что запускает контроллер DMA на чтение кодированных данных из CENC. Если после этого еще имеются блоки для кодирования (зависит от BLK_CNT), выполняется переход к шагу 3.
5. После обработки и чтения всех (BLK_CNT + 1) блоков контроллер DMA генерирует прерывание в CPU.
6. Если имеется очередной набор блоков для кодирования, то выполняется переход к шагу 2.

Далее приводится более подробное описание этих шагов.

22.3.3.2 Работа с DMA

Настройка контроллера DMA заключается в подготовке дескрипторов задач для пересылки необходимой информации.

В направлении TX выполняются следующие пересылки:

- конфигурационные параметры (регистры **CENC_BLKCFG***). Необходимо выполнять запись всех регистров, даже если их значения не изменялись;

- один или несколько блоков входных данных.

В направлении RX выполняется пересылка одного или нескольких блоков выходных данных.

Блоки входных/выходных данных записываются/считываются частями (порциями). Размер части определяется размером входного/выходного буфера FIFO и составляет 16 слов по 64 бита (1024 бит). Размер последней части блока может быть меньше.

Для уведомления контроллера DMA о возможности выполнить пересылку одной части блока данных CENC использует сигналы TXEVT (направление TX) и RXEVT (направление RX), подключенные к каналам DMA следующим образом (см. Раздел 6.4.5 «Распределение событий»):

- TXEVT – DMA0 канал 5;
- RXEVT – DMA0 канал 4.

На Рисунок 22.16 показан пример последовательности передачи данных при кодировании двух блоков (BLK_CNT = 1) с идентичным набором параметров. Здесь CFG – конфигурационные параметры, DI #i.j – часть j блока входных данных i, DO #i.j – часть j блока выходных данных i.

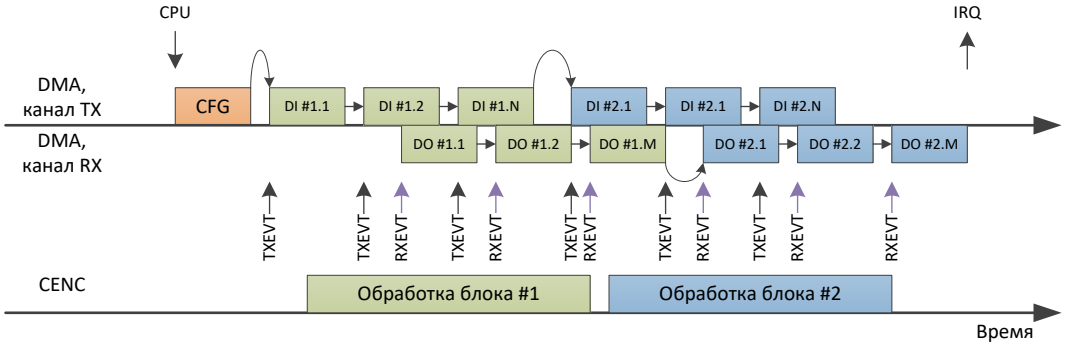


Рисунок 22.16. Пример кодирования двух блоков

Для ускорения передачи данных используется возможность DMA объединять несколько пересылок в цепочки. Так, в примере выше пересылка конфигурационных параметров связана в цепочку с пересылкой первого блока входных данных, который, в свою очередь, связан в цепочку с пересылкой второго блока входных данных.

В оптимальные значения параметров дескрипторов DMA для всех рассмотренных пересылок. В таблице используются следующие обозначения:

- N_{IN} – размер исходного сообщения в битах
- N_{OUT} – размер кодированного блока в битах
- ceil(x) – операция округления x до большего целого

Таблица 22.24 приведены оптимальные значения параметров дескрипторов DMA для всех рассмотренных пересылок. В таблице используются следующие обозначения:

- N_{IN} – размер исходного сообщения в битах
- N_{OUT} – размер кодированного блока в битах
- ceil(x) – операция округления x до большего целого

Таблица 22.24. Настройки контроллера DMA для блока CENC

Параметр DMA	Значения для соответствующего типа дескриптора		
	Конфигурация	Входные данные	Выходные данные
CSR.RUN	1	1	1
CSR.DIR	0	0	1
CSR.REV	0	0	0
CSR.2D	0	0	0
CSR.WN	7	7	7
CSR.SIZE	4	4	4
CSR.IM	0	0	0
CSR.CHEN	1	1	1
CSR.MASK	0	1	1
IR0	Адрес конфигурационных параметров в системной памяти	Адрес первого блока в системной памяти	Адрес первого блока выходных данных в системной памяти
IR1	Адрес регистра CENC_BLKCFG0	Адрес CENC_IBUF	Адрес CENC_OBUF
OR0	1	1	1
OR1	1	0	0
ORY	0 (не используется)	0	0
WCX	1	ceil[N_{IN} / 128] - 1	ceil[N_{OUT} / 128] - 1
WCY	0 (не используется)	0	0
CP	Адрес дескриптора блока входных данных	Адрес следующего дескриптора	Адрес следующего дескриптора

Примечание: Контроллер DMA оперирует 128-ми разрядными словами, а интерфейс CENC имеет разрядность 64 бита. Это накладывает дополнительные ограничения на конфигурацию DMA – если количество 64-х разрядных слов записываемых во входной буфер CENC или считываемых из выходного буфера нечетное, то контроллер DMA должен пересылать последнее слово с размером 64 бита вместо 128-ми (CSR.SIZE = 3).

Кодирование нового блока (или набора из нескольких блоков) может быть запущено после считывания данных предыдущего кодирования из CENC_OBUF.

22.3.3.3 Генерирующие полиномы

Генерирующие полиномы задаются аналогично VITDEC (см. п. 22.2.3.1).

22.3.3.4 Перфорация

Параметрыперфорации задаются аналогично VITDE (см. п. 22.2.3.4).

22.3.3.5 Формат входных/выходных данных

Блок CENC поддерживает несколько форматов упаковки данных в системной памяти. Преобразование формата выполняется автоматически при загрузке и выгрузке данных. За

формат входных и выходных данных отвечают параметры IN_ORDER и OUT_ORDER регистра **CENC_BLKCFG4** соответственно.

На рис Рисунок 22.17 показаны варианты расположения бит данных в системной памяти в зависимости от значений параметров IN_ORDER и OUT_ORDER. Бит 0 является первым битом, передаваемым в блоке данных.

IN_/OUT_ORDER = 0
(исходный порядок)

msb							lsb	
127	120	119	112	111	104	103	96	Base + 0xC
95	88	87	80	79	72	71	64	Base + 0x8
63	56	55	48	47	40	39	32	Base + 0x4
31	24	23	16	15	8	7	0	Base + 0x0

IN_/OUT_ORDER = 2
(обратный порядок байт в слове)

msb							lsb	
103	96	111	104	119	112	127	120	Base + 0xC
71	64	79	72	87	80	95	88	Base + 0x8
39	32	47	40	55	48	63	56	Base + 0x4
7	0	15	8	23	16	31	24	Base + 0x0

IN_/OUT_ORDER = 1
(обратный порядок бит в байте)

msb							lsb	
120	127	112	119	104	111	96	103	Base + 0xC
88	95	80	87	72	79	64	71	Base + 0x8
56	63	48	55	40	47	32	39	Base + 0x4
24	31	16	23	8	15	0	7	Base + 0x0

IN_/OUT_ORDER = 3
(обратный порядок бит и байт)

msb							lsb	
96	103	104	111	112	119	120	127	Base + 0xC
64	71	72	79	80	87	88	95	Base + 0x8
32	39	40	47	48	55	56	63	Base + 0x4
0	7	8	15	16	23	24	31	Base + 0x0

Рисунок 22.17. Формат входных/выходных данных CENC

22.3.3.6 Программный сброс

При необходимости, программный сброс выполняется путем записи значения 1 в регистр **CENC_SWRST**. Программный сброс приводит CENC в начальное состояние и не влияет на состояние регистров **CENC_BLKCFG***. Во время выполнения сброса все каналы DMA, работающие с CENC, должны быть остановлены, а также не должно быть активных транзакций на интерфейсе DMA_IF.

22.3.3.7 Прерывания

Аппаратные сигналы TXEVT, RXEVT также могут генерировать прерывание. Эту возможность можно использовать для отладки или для альтернативного (более медленного) способа работы с CENC без участия контроллера DMA.

Для использования прерываний необходимо установить соответствующие разряды в регистре **CENC_IRQ_EN**. Состояние разрешенных прерываний отображается в регистре **CENC_IRQ_EN_STAT**. Для сброса активного прерывания необходимо записать '1' в соответствующий разряд регистра **CENC_IRQ_CLR**.

22.3.4 Регистры

22.3.4.1 Перечень регистров

В Таблица 22.25 перечислены общие регистры статуса и управления CENC. Доступ к регистрам возможен через интерфейс CFG_IF только 32-х разрядными словами.

Таблица 22.25. Регистры статуса и управления CENC (интерфейс CFG_IF)

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
CENC_REVID	Версия RTL	0x13[1]	RO	0x0000
CENC_SWRST	Регистр программного сброса	0x00	WO	0x0004
CENC_STAT	Регистр статуса	0x31	RO	0x000C
CENC_TXES	Регистр установки события TXEVT	0x00	WO	0x0010
CENC_RXES	Регистр установки события RXEVT	0x00	WO	0x0014
CENC_IRQ_EN	Регистр разрешения прерываний	0x0	RW	0x0018
CENC_IRQ_STAT	Регистр статуса/установки прерываний	0x0	RW	0x001C
CENC_IRQ_EN_STAT	Регистр статуса разрешенных прерываний	0x0	RO	0x0020
CENC_IRQ_CLR	Регистр сброса прерываний	0x0	WO	0x0024

Примечания:

2) Версия RTL имеет формат 0xXY и соответствует версии SVN релиза. Например, для релиза r1p2 X = 1, Y = 2, значение регистра = 0x12

В Таблица 22.26 перечислены регистры для настройки параметров кодирования. Доступ к регистрам производится по интерфейсу DMA_IF. Поддерживаются 32-х, 64-х и 128-ми разрядные обращения. Запись по неиспользуемым адресам игнорируется.

Таблица 22.26. Регистры настройки кодера CENC (интерфейс DMA_IF)

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
CENC_BLKCFG0	Конфигурационный регистр 0	0x0	RW	0x0000
CENC_BLKCFG1	Конфигурационный регистр 1	0x0	RW	0x0004
CENC_BLKCFG2	Конфигурационный регистр 2	0x0	RW	0x0008
CENC_BLKCFG3	Конфигурационный регистр 3	0x0	RW	0x000C
CENC_BLKCFG4	Конфигурационный регистр 4	0x0	RW	0x0010
CENC_IBUF	Регистр записи буфера FIFO входных данных	0x0	WO	0x0080
CENC_OBUF	Регистр чтения буфера FIFO выходных данных	0x0	RO	0x00C0

22.3.4.2 Регистры статуса и управления

22.3.4.2.1 Регистр CENC_REVID

Таблица 22.27. Формат регистра CENC_REVID

Номер бита	Условное обозначение	Назначение
31:8	-	Резерв
7:4	X	Старшая часть номера версии (X)
3:0	Y	Младшая часть номера версии (Y)

22.3.4.2.2 Регистр CENC_SWRST

Таблица 22.28. Формат регистра CENC_SWRST

Номер бита	Условное обозначение	Назначение
31:1	-	Резерв
0	RESET	Запись 1 в этот бит инициирует программный сброс блока. Сброс не влияет на состояние регистров **CENC_BLKCFG**

22.3.4.2.3 Регистр CENC_STAT

Таблица 22.29. Формат регистра CENC_STAT

Номер бита	Условное обозначение	Назначение
31:24	-	Резерв
23:16	RX_BLOCK_CNT	Количество отправленных блоков
15:8	TX_BLOCK_CNT	Количество принятых блоков
7:6	OBUF_BUSY	Статус наполненности буферов FIFO выходных данных: 0: буфер не полон; 1: буфер полон
5:4	IBUF_EMPTY	Статус наполненности буферов FIFO входных данных: 0: буфер не пуст; 1: буфер пуст
3:2	-	Резерв
1	WAIT_TX_DATA	Значение 1 этого бита означает, что кодер находится в состоянии ожидания загрузки входных данных
0	WAIT_BLKCFG	Значение 1 этого бита означает, что кодер находится в состоянии ожидания записи регистров конфигурации **CENC_BLKCFG**

22.3.4.2.4 Регистр CENC_TXES

Таблица 22.30. Формат регистра CENC_TXES

Номер бита	Условное обозначение	Назначение
31:1	-	Резерв
0	TXEVT	Запись 1 в этот бит генерирует событие TXEVT

22.3.4.2.5 Регистр CENC_RXES

Таблица 22.31. Формат регистра CENC_RXES

Номер бита	Условное обозначение	Назначение
31:1	-	Резерв
0	RXEVT	Запись 1 в этот бит генерирует событие RXEVT

22.3.4.2.6 Регистр CENC_IRQ_EN

Таблица 22.32. Формат регистра CENC_IRQ_EN

Номер бита	Условное обозначение	Назначение
31:2	-	Резерв
1	RXEVT	Разрешение прерывания по событию RXEVT
0	TXEVT	Разрешение прерывания по событию TXEVT

22.3.4.2.7 Регистр CENC_IRQ_STAT

Регистр содержит биты статуса для всех прерываний. Бит статуса устанавливается в 1 при возникновении прерывания без учета разрешений, либо при записи 1 в этот бит (эта возможность может быть использована для отладочных целей). Запись 0 игнорируется.

Таблица 22.33. Формат регистра CENC_IRQ_STAT

Номер бита	Условное обозначение	Назначение
31:2	-	Резерв
1	RXEVT	Статус прерывания по событию RXEVT
0	TXEVT	Статус прерывания по событию TXEVT

22.3.4.2.8 Регистр CENC_IRQ_EN_STAT

Регистр содержит биты статуса для всех разрешенных прерываний. Бит статуса устанавливается в 1 при возникновении прерывания только, если установлен соответствующий бит в регистре **CENC_IRQ_EN**. Регистр доступен только для чтения.

Таблица 22.34. Формат регистра CENC_IRQ_EN_STAT

Номер бита	Условное обозначение	Назначение
31:2	-	Резерв
1	RXEVT	Статус прерывания по событию RXEVT
0	TXEVT	Статус прерывания по событию TXEVT

22.3.4.2.9 Регистр CENC_IRQ_CLR

Регистр предназначен для программного сброса прерываний, бит статуса которых установлен в регистре **CENC_IRQ_STAT**. Сброс производится путем записи 1 в соответствующий бит. Запись 0 игнорируется. При чтении регистра возвращается 0.

Таблица 22.35. Формат регистра CENC_IRQ_CLR

Номер бита	Условное обозначение	Назначение
31:2	-	Резерв
1	RXEVT	Сброс прерывания по событию RXEVT
0	TXEVT	Сброс прерывания по событию TXEVT

22.3.4.3 Регистры конфигурации кодового блока

22.3.4.3.1 Регистр CENC_BLKCFG0

Таблица 22.36. Формат регистра CENC_BLKCFG0

Номер бита	Условное обозначение	Назначение
31:24	POLY3	Генерирующий полином G_3
23:16	POLY2	Генерирующий полином G_2
15:8	POLY1	Генерирующий полином G_1
7:0	POLY0	Генерирующий полином G_0

22.3.4.3.2 Регистр CENC_BLKCFG1

Таблица 22.37. Формат регистра CENC_BLKCFG1

Номер бита	Условное обозначение	Назначение
31:24	POLY7	Генерирующий полином G_7
23:16	POLY6	Генерирующий полином G_6
15:8	POLY5	Генерирующий полином G_5
7:0	POLY4	Генерирующий полином G_4

22.3.4.3.3 Регистр CENC_BLKCFG2

Таблица 22.38. Формат регистра CENC_BLKCFG2

Номер бита	Условное обозначение	Назначение
31:24	PPAT3	Шаблон перфорации, строка 3
23:16	PPAT2	Шаблон перфорации, строка 2
15:8	PPAT1	Шаблон перфорации, строка 1
7:0	PPAT0	Шаблон перфорации, строка 0

22.3.4.3.4 Регистр CENC_BLKCFG3

Таблица 22.39. Формат регистра CENC_BLKCFG3

Номер бита	Условное обозначение	Назначение
31:24	PPAT7	Шаблон перфорации, строка 7
23:16	PPAT6	Шаблон перфорации, строка 6
15:8	PPAT5	Шаблон перфорации, строка 5
7:0	PPAT4	Шаблон перфорации, строка 4

22.3.4.3.5 Регистр CENC_BLKCFG4

Таблица 22.40. Формат регистра CENC_BLKCFG4

Номер бита	Условное обозначение	Назначение
31:28	-	Резерв
27:26	OUT_ORDER	Порядок бит выходных данных: 0: исходный порядок бит; 1: обратный порядок бит в каждом байте; 2: исходный порядок бит, обратный порядок байт в каждом слове; 3: обратный порядок бит в байте, обратный порядок байт в слове
25:24	IN_ORDER	Порядок бит входных данных: 0: исходный порядок бит; 1: обратный порядок бит в каждом байте; 2: исходный порядок бит, обратный порядок байт в каждом слове; 3: обратный порядок бит в байте, обратный порядок байт в слове
23:16	BLK_CNT	Количество блоков в наборе (значение минус 1)
15:0	BLK_LEN	Размер исходного сообщения в битах (значение минус 1)

22.4 Турбо декодер (TDC)

22.4.1 Общая информация

22.4.1.1 Основные особенности

Блок TDC (Turbo Decoder Coprocessor) предназначен для декодирования турбо-кодов, полученных путем параллельной конкатенации рекурсивных систематических сверточных кодов. Входными данными для TDC являются мягкие оценки переданных через канал систематических и паритетных бит. Выходными данными – жесткие решения.

TDC имеет следующие характеристики:

- Кодовая скорость: 1/3, 1/5, 1/7
- Алгоритм декодирования: Max-Log-MAP
- Размер декодируемого блока: 40 – 8192 бит
- Поддержка перемежения по конфигурируемой таблице
- Внутренний квадратурный перемежитель
- Программируемое количество итераций: 1 – 16
- Разрядность входных метрик: 6 бит
- Двухбуферная схема загрузки данных
- Аппаратный интерфейс для работы с DMA
- Производительность: 90 Мбит/с @ 600 МГц, 6 итераций

22.4.1.2 Структура кодера

TDC работает с турбо-кодом, полученными с помощью кодера, структура которого показана на Рисунок 22.18. Турбо кодер состоит из двух рекурсивных систематических сверточных кодеров (Recursive Systematic Convolutional Codes - RSC), соединенных параллельно, и перемежителя.

На вход первого RSC кодера поступает последовательность бит непосредственно со входа X. На вход второго RSC кодера данные поступают с выхода перемежителя (Interleaver), переставляющего биты исходного сообщения по заданному закону. Каждый составной кодер имеет кодовое ограничение 4 и четыре выхода: систематические данные (X_0, X_1), проверочные (паритетные) данные (A_0, A_1, B_0, B_1, C_0, C_1). Систематический выход X_1 используется только на завершающих стадиях треллиса для приведения кодера в нулевое состояние.

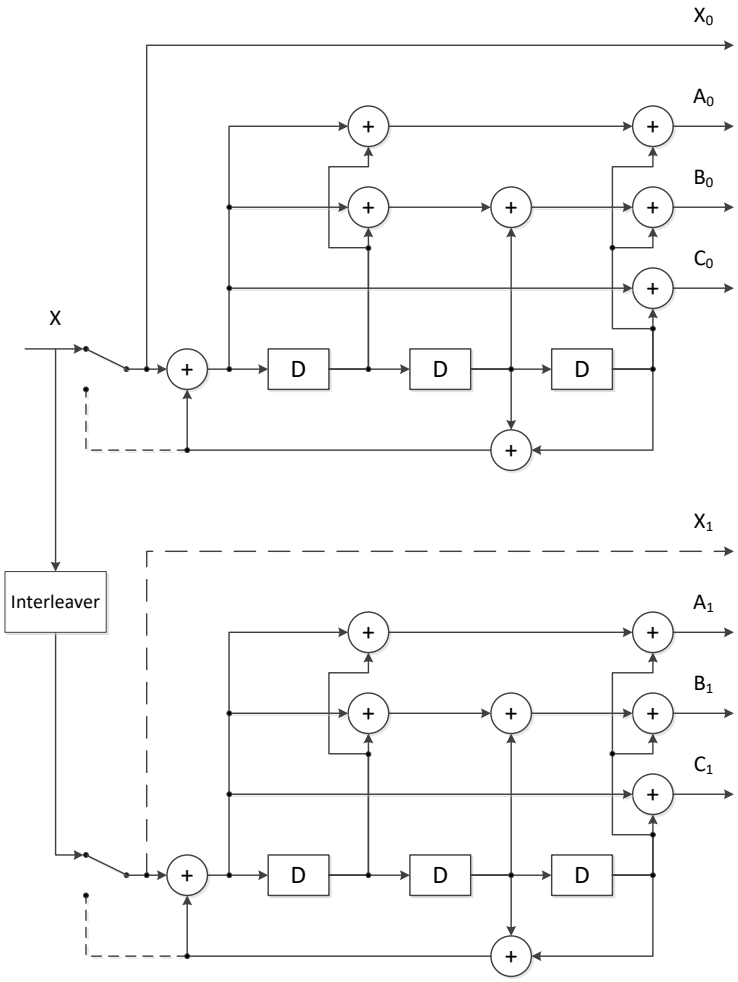


Рисунок 22.18. Структура кодера

Перед кодированием блока данных оба составных кодера находятся в нулевом состоянии. После кодирования блока оба кодера также переводятся в нулевое состояние. Это осуществляется с помощью переключателей, показанных на Рисунок 22.18. В течение первых трех тактов переключатель первого RSC кодера находится в нижнем положении, и на выход передаются биты X_0, A_0, B_0, C_0 («хвостовые» биты). В течение следующих трех тактов переключатель второго RSC кодера находится в нижнем положении, и на выход передаются биты X_1, A_1, B_1, C_1.

На выходе турбо кодера выполняется перфорация данных для получения необходимой кодовой скорости. Возможны следующие варианты выходной последовательности бит:

- кодовая скорость 1/3: $X_0[0]$, $A_0[0]$, $A_1[0]$, $X_0[1]$, $A_0[1]$, $A_1[1]$, ...
- кодовая скорость 1/5: $X_0[0]$, $A_0[0]$, $B_0[0]$, $A_1[0]$, $B_1[0]$, $X_0[1]$, $A_0[1]$, $B_0[1]$, $A_1[1]$, $B_1[1]$, ...
- кодовая скорость 1/7: $X_0[0]$, $A_0[0]$, $B_0[0]$, $C_0[0]$, $A_1[0]$, $B_1[0]$, $C_1[0]$, $X_0[1]$, $A_0[1]$, $B_0[1]$, $C_0[1]$, $A_1[1]$, $B_1[1]$, $C_1[1]$, ...

22.4.2 Структурная схема

22.4.2.1 Структурная схема TDC

Структурная схема турбо декодера показана на Рисунок 22.19.

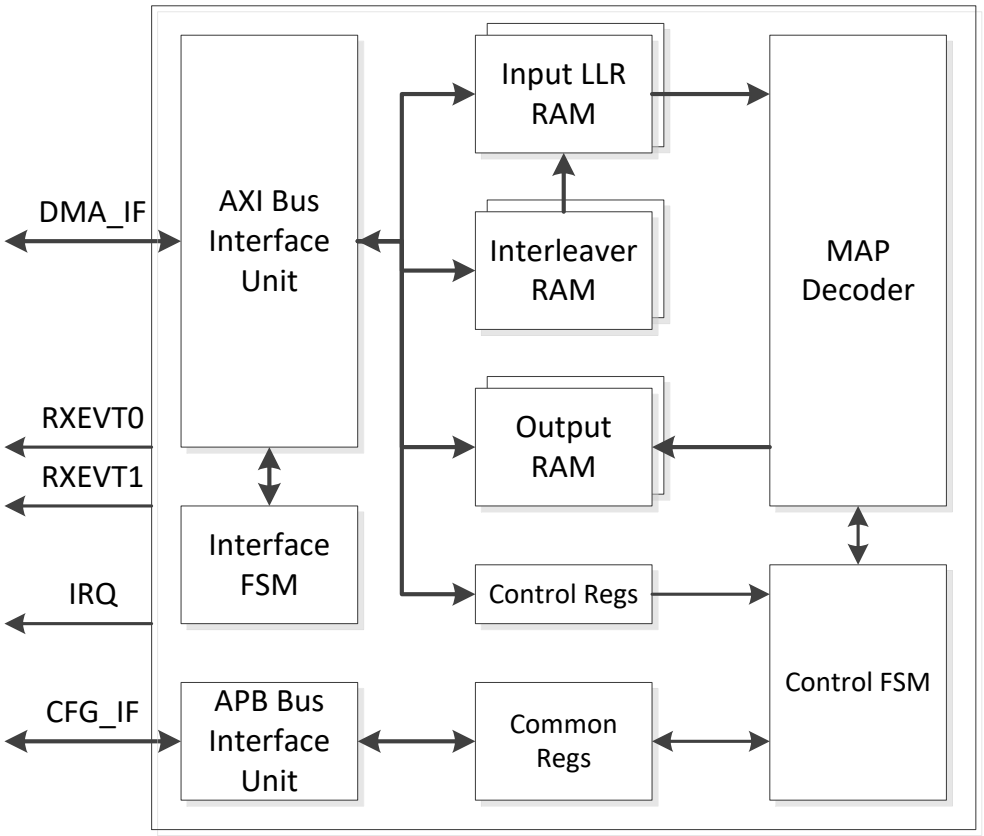


Рисунок 22.19. Структурная схема турбо декодера

Декодер состоит из следующих основных блоков:

- AXI Bus Interface Unit: блок реализует slave-устройство для передачи данных по интерфейсу DMA_IF
- Interface FSM: блок управляет загрузкой входных данных, запуском процесса декодирования, выгрузкой результатов декодирования
- APB Bus Interface Unit: блок реализует slave-устройство для передачи данных по интерфейсу CFG_IF
- Input LLR RAM: блоки памяти для хранения входных метрик
- Interleaver RAM: блоки памяти для хранения таблицы перемежения
- Output RAM: блоки памяти для хранения результатов декодирования
- Control Regs: регистры, содержащие параметры для декодирования текущего блока данных

- Common Regs: общие регистры статуса, настройки прерываний, а также глобальных настроек декодера
- MAP Decoder: ядро декодера по алгоритму максимума апостериорной вероятности
- Control FSM: блок управления MAP-декодером.

22.4.2.2 Interface FSM

Блок Interface FSM предназначен для контроля загрузки входных данных и параметров, запуска декодирования и контроля выгрузки результатов декодирования.

Диаграмма состояний блока показана на Рисунок 22.20.

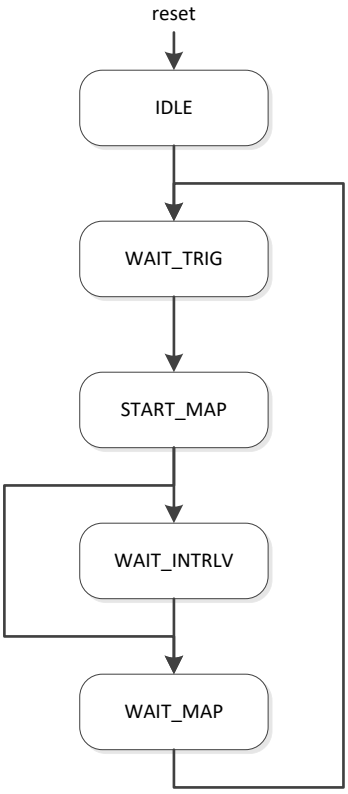


Рисунок 22.20. Диаграмма состояний блока Interface FSM

После аппаратного или программного сброса блок находится в состоянии IDLE в течение одного такта (в текущей реализации в данном состоянии не выполняется никаких действий), после чего переходит в состояние ожидания загрузки конфигурации и запуска декодера (WAIT_TRIG).

После записи единицы в регистр TRIG одного из буферов, в соответствующем регистре **TDC_STAT_B0/B1** устанавливается бит PEND. При обнаружении установленного бита PEND у одного из буферов блок Interface FSM устанавливает бит BUSY у соответствующего буфера и производит запуск декодера (состояние START_MAP).

При установленном бите INTRL_LOAD регистра **TDC_BLKCFG2** блок ожидает загрузку таблицы перемежения (состояние WAIT_INTRLV), после чего переходит в состояние WAIT_MAP, в котором дожидается завершения декодирования. Если декодирование первой половины итерации завершается раньше загрузки таблицы перемежения, то MAP-декодер приостанавливает работу до выхода из состояния WAIT_INTRLV. Если загрузка таблицы перемежения не требуется (бит INTRL_LOAD сброшен), то состояние WAIT_INTRLV пропускается.

После завершения декодирования текущего буфера генерируется событие REVT0/1, и блок Interface FSM возвращается в состояние WAIT_TRIG.

22.4.2.3 MAP декодер

Структурная схема MAP декодера показана на Рисунок 22.21. На вход блока поступают оценки символов («мягкие» решения) систематической (Ls) и проверочных (Lp) частей кодового блока, а также априорная информация о переданном символе (La). На выходе блока формируется оценка переданного символа (Le), используемая в качестве априорной информации для последующих итераций декодирования.

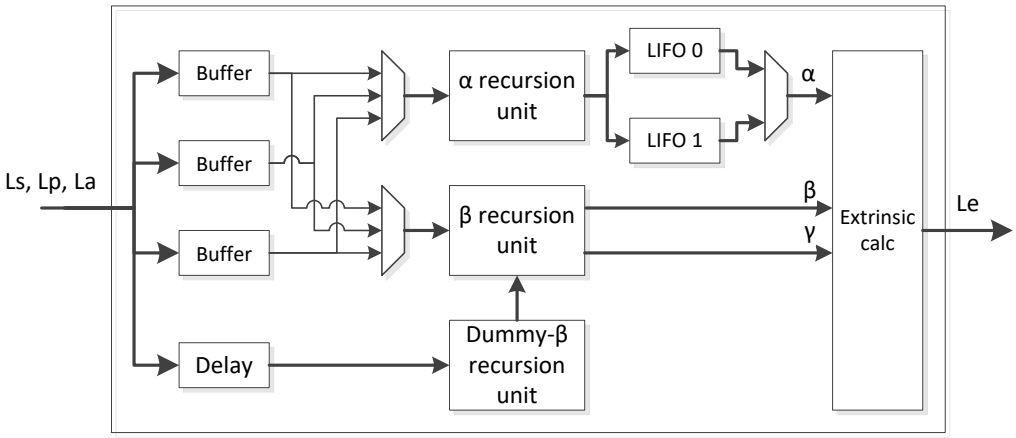


Рисунок 22.21. Структурная схема MAP декодера

В процессе декодирования вычисляются промежуточные метрики α, β, γ. Для уменьшения объемов памяти, необходимой для хранения промежуточных значений, используется принцип скользящего окна (sliding window). Блок данных разбивается на фрагменты фиксированного размера по L символов. Одновременно производится обработка трех фрагментов (окон, Рисунок 22.22), при этом над каждым фрагментом выполняется одна из трех операций:

1. загрузка фрагмента (w + 1) в буфер и вычисление по нему обратных метрик состояний β, служащих для инициализации окна w;
2. вычисление прямых метрик состояний α по фрагменту w;
3. вычисление обратных (β) и апостериорных метрик (Le) по фрагменту (w - 1).

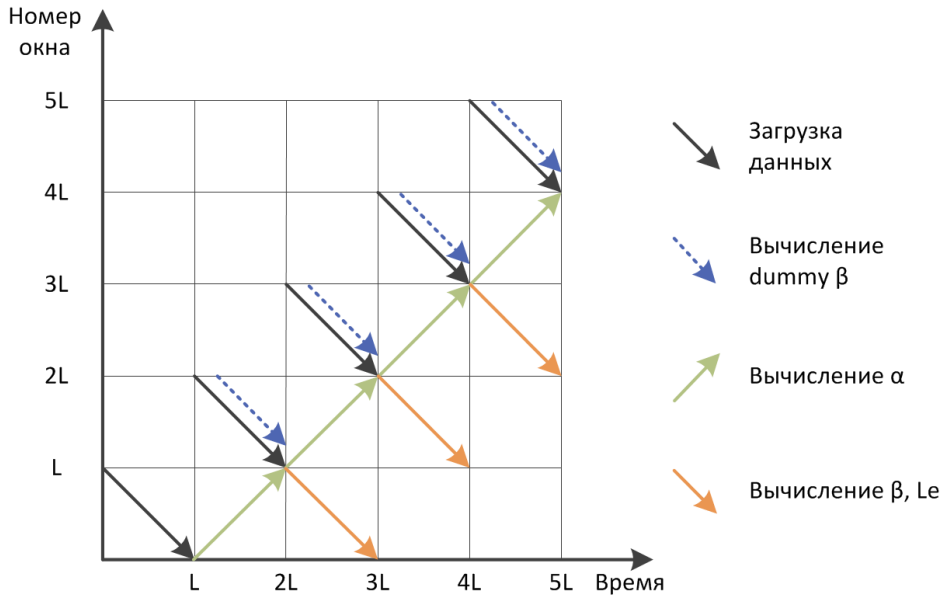


Рисунок 22.22. Порядок вычисления метрик

22.4.3 Функционирование

22.4.3.1 Введение

Общий алгоритм использования блока турбо декодера состоит из следующих шагов:

1. Настройка хост-процессором (CPU) глобальных параметров декодера через интерфейс CFG_IF.
2. Подготовка с помощью CPU входных данных и параметров для декодирования:
   - вычисление входных метрик;
   - генерация таблицы перемежения (если используется внешняя таблица);
   - вычисление параметров декодера (размер окна, начальные значения β).
3. Настройка блока DMA для передачи входных метрик, индексов перемежения и параметров декодера. Необходимо выполнять запись всех параметров, даже если их значения не изменялись.
4. Запуск процесса декодирования путем записи в регистр **TDC_TRIG**.
5. По завершению декодирования турбо декодер генерирует событие RXEVT.
6. Хост-процессор, обнаружив событие RXEVT, запускает блок DMA для передачи результатов декодирования.
7. После чтения результатов декодирования блок DMA генерирует прерывание для CPU.

22.4.3.2 Работа с DMA

Процесс обработки одного блока данных показан на Рисунок 22.23. Здесь используются следующие обозначения:

- **Start DMA**: запуск цепочки задач блока DMA;
- **Config**, **Input LLR**, **Interl. Tbl**: загрузка конфигурации, входных метрик и таблицы перемежения в декодер;
- **T**: запуск декодирования с помощью записи в регистр TRIG;
- **Decoding**: процесс декодирования;
- **HD**: чтение декодированных данных;
- **ISR**: процедура обработки прерывания в CPU.

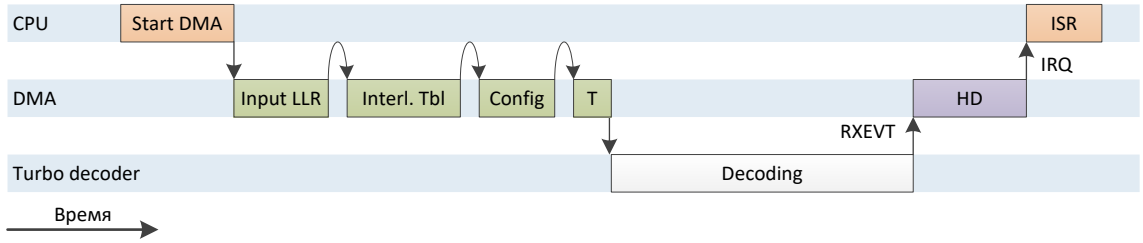


Рисунок 22.23. Общий процесс декодирования одного блока

Для ускорения передачи данных используется возможность DMA объединять несколько пересылок в цепочки. Так, в примере выше в одну цепочку связаны загрузка входных данных, таблицы и конфигурационных параметров.

В декодере реализована двухбуферная схема работы: есть два идентичных набора параметров декодера, буферов входных и выходных данных, условно обозначаемых как B0, B1. Данная схема позволяет выполнять загрузку следующего блока данных во время декодирования текущего блока, что сокращает общую задержку на обработку следующего блока (Рисунок 22.24). При этом параметры декодирования (например, размер блока, количество итераций), а также таблицы перемежения, загружаемые в разные буферы, могут отличаться. Таким образом, можно обрабатывать независимые потоки данных с различными параметрами.

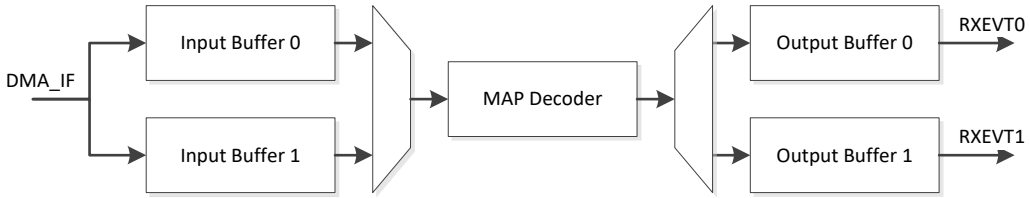


Рисунок 22.24. Двухбуферная схема

Для уведомления контроллера DMA о возможности считать декодированные данные из памяти TDC используются сигналы RXEVT0 и RXEVT1 для буферов B0 и B1 соответственно. Сигналы подключены к каналам DMA следующим образом (см. Раздел 6.4.5 «Распределение событий»):

- RXEVT0 – DMA1 канал 7;
- RXEVT1 – DMA1 канал 6.

На Рисунок 22.25 показан пример взаимодействия декодера с CPU и DMA в двухбуферном режиме.

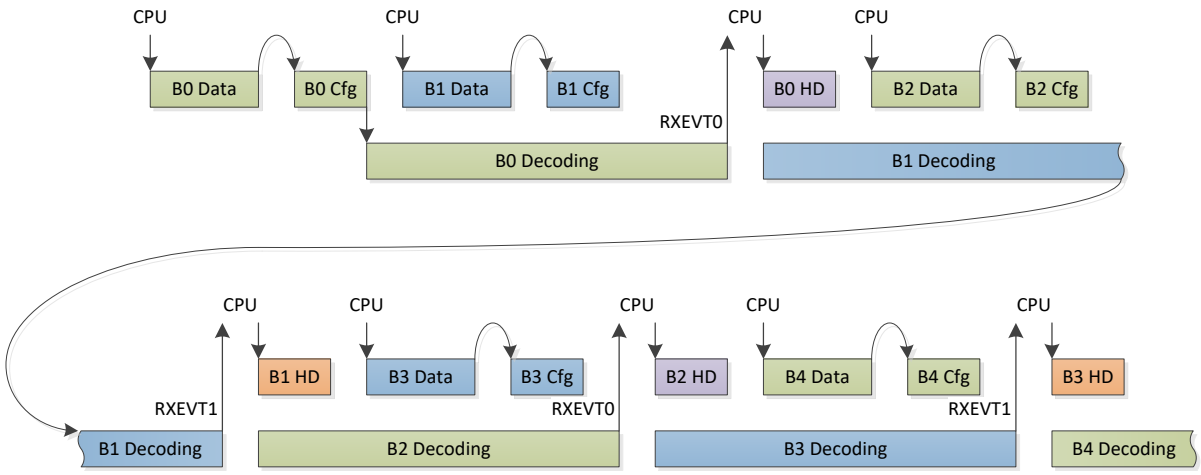


Рисунок 22.25. Декодирование в двухбуферном режиме

Предупреждение: во время процесса декодирования (с момента записи регистра **TDC_TRIG** до события RXEVT) все обращения к памяти, а также запись регистров **TDC_BLKCFG*** обрабатываемого буфера запрещены и возвращают статус ошибки на интерфейсе AXI.

22.4.3.3 Формат входных данных

Метрики логарифмического отношения правдоподобия (LLR), получаемые в DSP, имеют разрядность 6 бит. Каждое значение LLR в памяти декодера хранится в одном байте, при этом старшие два бита не используются декодером. В целочисленном формате каждое значение LLR имеет диапазон от -32 до 31. Также метрики могут быть представлены в дробном формате, при этом положение точки для декодера не имеет значения.

22.4.3.4 Порядок входных данных в памяти декодера

22.4.3.4.1 Размер блока кратен 2

В следующих таблицах показан порядок расположения кодированных данных в памяти TDC в зависимости от скорости кодирования, если размер блока K кратен 2. Банк памяти SYST предназначен для систематических данных, банки PAR0 - PAR5 – для паритетных данных.

Таблица 22.41. Расположение кодированных данных в памяти для скорости 1/3

Смещение от базового адреса	SYST	PAR0	PAR1	PAR2	PAR3	PAR4	PAR5
0	$X_0[0]$	$A_0[0]$	-	-	$A_1[0]$	-	-
1	$X_0[1]$	$A_0[1]$	-	-	$A_1[1]$	-	-
…	…	…	…	…	…	…	…
K – 1	$X_0[K - 1]$	$A_0[K - 1]$	-	-	$A_1[K - 1]$	-	-

Таблица 22.42. Расположение кодированных данных в памяти для скорости 1/5

Смещение от базового адреса	SYST	PAR0	PAR1	PAR2	PAR3	PAR4	PAR5
0	$X_0[0]$	$A_0[0]$	$B_0[0]$	-	$A_1[0]$	$B_1[0]$	-
1	$X_0[1]$	$A_0[1]$	$B_0[1]$	-	$A_1[1]$	$B_1[1]$	-
…	…	…	…	…	…	…	…
K – 1	$X_0[K - 1]$	$A_0[K - 1]$	$B_0[K - 1]$	-	$A_1[K - 1]$	$B_1[K - 1]$	-

Таблица 22.43. Расположение кодированных данных в памяти для скорости 1/7

Смещение от базового адреса	SYST	PAR0	PAR1	PAR2	PAR3	PAR4	PAR5
0	$X_0[0]$	$A_0[0]$	$B_0[0]$	$C_0[0]$	$A_1[0]$	$B_1[0]$	$C_1[0]$
1	$X_0[1]$	$A_0[1]$	$B_0[1]$	$C_0[1]$	$A_1[1]$	$B_1[1]$	$C_1[1]$
…	…	…	…	…	…	…	…
K – 1	$X_0[K - 1]$	$A_0[K - 1]$	$B_0[K - 1]$	$C_0[K - 1]$	$A_1[K - 1]$	$B_1[K - 1]$	$C_1[K - 1]$

22.4.3.4.2 Размер блока не кратен 2

Если размер блока не кратен 2, то входные данные должны быть расширены с использованием значений LLR, соответствующим завершающим битам.

В следующих таблицах показан порядок расположения кодированных данных в памяти TDC в зависимости от скорости кодирования, если размер блока (K) не кратен 2.

$Xt_i[k]$ - значение LLR для завершающего бита систематического выхода кодера i;
$At_i[k]$ - значение LLR для завершающего бита проверочного выхода A кодера i;
$Bt_i[k]$ - значение LLR для завершающего бита проверочного выхода B кодера i;
$Ct_i[k]$ - значение LLR для завершающего бита проверочного выхода C кодера i.

Таблица 22.44. Расположение кодированных данных в памяти для скорости 1/3

Смещение от базового адреса	SYST	PAR0	PAR1	PAR2	PAR3	PAR4	PAR5
0	$X_0[0]$	$A_0[0]$	-	-	$A_1[0]$	-	-
1	$X_0[1]$	$A_0[1]$	-	-	$A_1[1]$	-	-
…	…	…	…	…	…	…	…
K – 1	$X_0[K - 1]$	$A_0[K - 1]$	-	-	$A_1[K - 1]$	-	-
K	$Xt_0[0]$	$At_0[0]$	-	-	$At_1[0]$	-	-

Таблица 22.45. Расположение кодированных данных в памяти для скорости 1/5

Смещение от базового адреса	SYST	PAR0	PAR1	PAR2	PAR3	PAR4	PAR5
0	$X_0[0]$	$A_0[0]$	$B_0[0]$	-	$A_1[0]$	$B_1[0]$	-
1	$X_0[1]$	$A_0[1]$	$B_0[1]$	-	$A_1[1]$	$B_1[1]$	-
…	…	…	…	…	…	…	…
K – 1	$X_0[K - 1]$	$A_0[K - 1]$	$B_0[K - 1]$	-	$A_1[K - 1]$	$B_1[K - 1]$	-
K	$Xt_0[0]$	$At_0[0]$	$Bt_0[0]$	-	$At_1[0]$	$Bt_1[0]$	-

Таблица 22.46. Расположение кодированных данных в памяти для скорости 1/7

Смещение от базового адреса	SYST	PAR0	PAR1	PAR2	PAR3	PAR4	PAR5
0	$X_0[0]$	$A_0[0]$	$B_0[0]$	$C_0[0]$	$A_1[0]$	$B_1[0]$	$C_1[0]$
1	$X_0[1]$	$A_0[1]$	$B_0[1]$	$C_0[1]$	$A_1[1]$	$B_1[1]$	$C_1[1]$
…	…	…	…	…	…	…	…
K – 1	$X_0[K - 1]$	$A_0[K - 1]$	$B_0[K - 1]$	$C_0[K - 1]$	$A_1[K - 1]$	$B_1[K - 1]$	$C_1[K - 1]$
K	$Xt_0[0]$	$At_0[0]$	$Bt_0[0]$	$Ct_0[0]$	$At_1[0]$	$Bt_1[0]$	$Ct_1[0]$

22.4.3.5 Таблица перемежения

Таблица перемежения может генерироваться внутри декодера либо храниться во внешней (по отношению к TDC) памяти и загружаться с помощью DMA.

22.4.3.5.1 Внутренняя генерация таблицы перемежения

TDC имеет встроенный генератор таблицы перемежения, включаемый при установке параметра ITG_MODE = 2 (регистр **TDC_BLKCFG2**). Закон перемежения соответствует стандарту LTE и задается следующей формулой:

$$p(i) = (f_1 * i + f_2 * i^2) \bmod K,$$

где f_1, f_2 – постоянные параметры; K – размер кодового блока; i - порядковый индекс до перемежения; p(i) – порядковый индекс после перемежения; mod – операция деления по модулю.

Для работы генератора предварительно необходимо рассчитать несколько параметров, перечисленных в Таблица 22.47.

Таблица 22.47. Параметры внутреннего генератора таблицы перемежения

Регистр	Поле	Значение
CFG10	ITG_P0	$(2 * f_2) \bmod K$
CFG10	ITG_P1	$(f_1 + f_2) \bmod K$

Значения f_1, f_2 определяются в зависимости от размера блока K согласно Таблица 22.48. В таблице приведены значения параметров f_1, f_2 для $F_{max} = 6144$. При необходимости, значения f_1, f_2 могут быть получены для $F > 6144$ (но не более 8192).

Таблица 22.48. Параметры f_1, f_2 перемежителя

i	K	f_1	f_2	i	K	f_1	f_2	i	K	f_1	f_2	i	K	f_1	f_2
1	40	3	10	48	416	25	52	95	1120	67	140	142	3200	111	240
2	48	7	12	49	424	51	106	96	1152	35	72	143	3264	443	204
3	56	19	42	50	432	47	72	97	1184	19	74	144	3328	51	104
4	64	7	16	51	440	91	110	98	1216	39	76	145	3392	51	212
5	72	7	18	52	448	29	168	99	1248	19	78	146	3456	451	192
6	80	11	20	53	456	29	114	100	1280	199	240	147	3520	257	220
7	88	5	22	54	464	247	58	101	1312	21	82	148	3584	57	336
8	96	11	24	55	472	29	118	102	1344	211	252	149	3648	313	228
9	104	7	26	56	480	89	180	103	1376	21	86	150	3712	271	232
10	112	41	84	57	488	91	122	104	1408	43	88	151	3776	179	236

i	K	f_1	f_2	i	K	f_1	f_2	i	K	f_1	f_2	i	K	f_1	f_2
11	120	103	90	58	496	157	62	105	1440	149	60	152	3840	331	120
12	128	15	32	59	504	55	84	106	1472	45	92	153	3904	363	244
13	136	9	34	60	512	31	64	107	1504	49	846	154	3968	375	248
14	144	17	108	61	528	17	66	108	1536	71	48	155	4032	127	168
15	152	9	38	62	544	35	68	109	1568	13	28	156	4096	31	64
16	160	21	120	63	560	227	420	110	1600	17	80	157	4160	33	130
17	168	101	84	64	576	65	96	111	1632	25	102	158	4224	43	264
18	176	21	44	65	592	19	74	112	1664	183	104	159	4288	33	134
19	184	57	46	66	608	37	76	113	1696	55	954	160	4352	477	408
20	192	23	48	67	624	41	234	114	1728	127	96	161	4416	35	138
21	200	13	50	68	640	39	80	115	1760	27	110	162	4480	233	280
22	208	27	52	69	656	185	82	116	1792	29	112	163	4544	357	142
23	216	11	36	70	672	43	252	117	1824	29	114	164	4608	337	480
24	224	27	56	71	688	21	86	118	1856	57	116	165	4672	37	146
25	232	85	58	72	704	155	44	119	1888	45	354	166	4736	71	444
26	240	29	60	73	720	79	120	120	1920	31	120	167	4800	71	120
27	248	33	62	74	736	139	92	121	1952	59	610	168	4864	37	152
28	256	15	32	75	752	23	94	122	1984	185	124	169	4928	39	462
29	264	17	198	76	768	217	48	123	2016	113	420	170	4992	127	234
30	272	33	68	77	784	25	98	124	2048	31	64	171	5056	39	158
31	280	103	210	78	800	17	80	125	2112	17	66	172	5120	39	80
32	288	19	36	79	816	127	102	126	2176	171	136	173	5184	31	96
33	296	19	74	80	832	25	52	127	2240	209	420	174	5248	113	902
34	304	37	76	81	848	239	106	128	2304	253	216	175	5312	41	166
35	312	19	78	82	864	17	48	129	2368	367	444	176	5376	251	336
36	320	21	120	83	880	137	110	130	2432	265	456	177	5440	43	170
37	328	21	82	84	896	215	112	131	2496	181	468	178	5504	21	86
38	336	115	84	85	912	29	114	132	2560	39	80	179	5568	43	174
39	344	193	86	86	928	15	58	133	2624	27	164	180	5632	45	176
40	352	21	44	87	944	147	118	134	2688	127	504	181	5696	45	178
41	360	133	90	88	960	29	60	135	2752	143	172	182	5760	161	120
42	368	81	46	89	976	59	122	136	2816	43	88	183	5824	89	182
43	376	45	94	90	992	65	124	137	2880	29	300	184	5888	323	184
44	384	23	48	91	1008	55	84	138	2944	45	92	185	5952	47	186
45	392	243	98	92	1024	31	64	139	3008	157	188	186	6016	23	94
46	400	151	40	93	1056	17	66	140	3072	47	96	187	6080	47	190
47	408	155	102	94	1088	171	204	141	3136	13	28	188	6144	263	480

22.4.3.5.2 Внешняя генерация таблицы перемежения

Таблица перемежения также может генерироваться с помощью DSP или храниться во внешней памяти, а затем загружаться в TDC с помощью DMA. Таблица представляет собой массив 13-ти битных целочисленных индексов. В памяти каждый индекс занимает два байта, при этом старшие 3 бита не используются.

При использовании внешней таблицы в общем случае закон перемежения может быть произвольным. При этом в параметр ITG_MODE необходимо записать значение 0. При такой конфигурации вычислительные возможности декодера не могут быть использованы полностью, что приводит к снижению производительности.

Если закон перемежения не изменяет четности индексов, т.е. четному (нечетному) индексу i до перемежения соответствует четный (нечетный) индекс $p(i)$ после перемежения ($i = 0..K$

- 1), то аппаратура декодера будет использоваться максимально эффективно. В этом случае параметр ITG_MODE = 1.

Если размер блока (K) не кратен 2, то в конец внешней таблицы перемежения должен быть добавлен 1 индекс со значением K.

22.4.3.6 Вычисление параметров скользящего окна

Для уменьшения объемов памяти, необходимой для хранения промежуточных значений, используется принцип скользящего окна (sliding window). Блок данных разбивается на фрагменты (окна). Размер окна влияет на производительность, а также на корректирующую способность декодера. Возможно наличие до трех окон с различными размерами, обозначаемыми как SW0, SW1, SW2. Блок может содержать несколько окон SW0, одно окно SW1, одно либо ни одного окна SW2. Максимальный размер окна - 128 символов. Размеры окон, а также количество окон SW0 вычисляются с помощью следующей процедуры.

```
Входные параметры:
  blockLen - исходный размер блока
  swNomLen - необходимый размер окна из списка {16, 32, 48, 64, 96, 128}
  Значение swNomLen выбирается так, чтобы выполнялось условие
    swNomLen >= blockLen/64
Процедура:
  step = 2;
  // Расширение размера блока до кратности step
  blockLenExt = step*ceil(blockLen/step);
  // Общее количество окон в блоке
  numSWTotal = ceil(blockLenExt / swNomLen);
  // Вычисление фактических размеров окон SW0, SW1, SW2 и количества окон SW0
  if (numSWTotal == 1) {
    sw0Len = 0;
    sw1Len = blockLenExt;
    sw2Len = 0;
    numSW0 = 0;
  } else if (numSWTotal = 2) {
    sw0Len = 0;
    sw1Len = step*ceil(blockLenExt/step/2);
    sw2Len = step*floor(blockLenExt/step/2);
    numSW0 = 0;
  } else if ((blockLenExt % swNomLen) <= swNomLen/2) {
    sw0Len = swNomLen;
    sw1Len = step*ceil((blockLenExt - (numSWTotal - 2)*swNomLen)/step/2);
    sw2Len = step*floor((blockLenExt - (numSWTotal - 2)*swNomLen)/step/2);
    numSW0 = numSWTotal - 2;
  } else {
    sw0Len = swNomLen;
    sw1Len = blockLenExt % swNomLen;
    sw2Len = 0;
    numSW0 = numSWTotal - 1;
  }
Обозначения:
```

```
ceil() - округление до целого к плюс бесконечности
floor() - округление до целого к нулю
% - деление по модулю
```

22.4.3.7 Вычисление начальных значений Beta

После кодирования блока данных кодеры приводятся к нулевому состоянию с помощью завершающих битов (tail bits). Соответствующие мягкие оценки LLR используются в декодере для вычисления начальных значений обратных метрик состояний (beta). Начальные значения beta вычисляются для прямой фазы декодирования (MAP0) и для перемеженной фазы (MAP1), и записываются в соответствующие регистры (**TDC_BLKCFG4** – **TDC_BLKCFG7**).

Значения β вычисляются в целочисленном знаковом формате. Используются следующие обозначения:

- $Xt_i[k]$ – значение LLR для завершающего бита систематического выхода кодера i;
- $At_i[k]$ – значение LLR для завершающего бита проверочного выхода A кодера i;
- $Bt_i[k]$ – значение LLR для завершающего бита проверочного выхода B кодера i;
- $Ct_i[k]$ – значение LLR для завершающего бита проверочного выхода C кодера i;
- sat() – ограничение в диапазоне [-128; 127];
- i = 0, 1 – номер составного сверточного кодера.

Если размер блока *K* кратен 2, то используются все три завершающих стадии треллиса: K, K + 1, K + 2. В этом случае вычисления выполняются по формулам:

```
βi[0] = 0
βi[1] = Xti[0] + Ati[0] + Bti[0] + Cti[0]
βi[2] = Xti[0] + Bti[0] + Xti[1] + Ati[1] + Bti[1] + Ct[1]
βi[3] = Ati[0] + Cti[0] + Xti[1] + Ati[1] + Bti[1] + Ct[1]
βi[4] = Ati[0] + Bti[0] + Xti[1] + Bti[1] +
        Xti[2] + Ati[2] + Bti[2] + Cti[2]
βi[5] = Xti[0] + Cti[0] + Xti[1] + Bti[1] +
        Xti[2] + Ati[2] + Bti[2] + Cti[2]
βi[6] = Xti[0] + Ati[0] + Ati[1] + Cti[1] +
        Xti[2] + Ati[2] + Bti[2] + Cti[2]
βi[7] = Bti[0] + Cti[0] + Ati[1] + Cti[1] +
        Xti[2] + Ati[2] + Bti[2] + Cti[2]

Нормировка и сатурация результата
βimax = max(βi[0], βi[1], βi[2], βi[3], βi[4], βi[5], βi[6], βi[7])

βi[0] = sat(βi[0] - βimax + 127)
βi[1] = sat(βi[1] - βimax + 127)
βi[2] = sat(βi[2] - βimax + 127)
βi[3] = sat(βi[3] - βimax + 127)
βi[4] = sat(βi[4] - βimax + 127)
βi[5] = sat(βi[5] - βimax + 127)
βi[6] = sat(βi[6] - βimax + 127)
βi[7] = sat(βi[7] - βimax + 127)
```

Если размер блока *K* не кратен 2, то используются две завершающих стадии треллиса и следующие формулы:

```
βi[0] = 0
βi[1] = Xti[1] + Ati[1] + Bti[1] + Cti[1]
βi[2] = Xti[1] + Bti[1] + Xti[2] + Ati[2] + Bti[2] + Cti[2]
βi[3] = Ati[1] + Cti[1] + Xti[2] + Ati[2] + Bti[2] + Cti[2]

Нормировка и сатурация результата
βimax = max(βi[0], βi[1], βi[2], βi[3])

βi[0] = sat(βi[0] - βimax + 127)
βi[1] = sat(βi[1] - βimax + 127)
βi[2] = sat(βi[2] - βimax + 127)
βi[3] = sat(βi[3] - βimax + 127)
βi[4] = -128
βi[5] = -128
βi[6] = -128
βi[7] = -128
```

Если завершающие биты не используются, и конечное состояние кодеров неизвестно, то

```
βi[j] = 0
j = 0, 1, ..., 7
```

22.4.4 Регистры

22.4.4.1 Перечень регистров и блоков памяти TDC

В Таблица 22.49 перечислены общие регистры статуса и управления TDC. Доступ к регистрам возможен через интерфейс CFG_IF только 32-х разрядными словами.

Таблица 22.49. Общие регистры статуса и управления TDC (CFG_IF)

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
TDC_REVID	Версия RTL	0x16[1]	RO	0x0000
TDC_SWRST	Программный сброс	0x0	WO	0x0004
TDC_STAT_B0	Регистр статуса декодера для буфера B0	0x0	RO	0x0010
TDC_EVSET_B0	Регистр установки события REVT для буфера B0	0x0	WO	0x0014
TDC_STAT_B1	Регистр статуса декодера для буфера B1	0x0	RO	0x0020
TDC_EVSET_B1	Регистр установки события REVT для буфера B1	0x0	WO	0x0024
TDC_IRQ_EN	Регистр разрешения прерываний	0x0	RW	0x0028
TDC_IRQ_STAT	Регистр статуса/установки прерываний	0x0	RW	0x002C
TDC_IRQ_EN_STAT	Регистр статуса разрешенных прерываний	0x0	RO	0x0030
TDC_IRQ_CLR	Регистр сброса прерываний	0x0	WO	0x0034

Примечания:
1) Версия RTL имеет формат 0xXY и соответствует версии SVN релиза. Например, для релиза r1p2 X = 1, Y = 2, значение регистра = 0x12

В Таблица 22.50 – Таблица 22.52 перечислены регистры для настройки параметров декодирования и блоки памяти TDC. Доступ к регистрам и памяти производится по интерфейсу DMA_IF. Поддерживаются 32-х, 64-х и 128-ми разрядные обращения к регистрам. Обращения к памяти могут быть с разрядностью 8, 16, 32, 64, 128 бит. Запись по неиспользуемым адресам игнорируется.

Таблица 22.50. Регистры статуса и настроек декодирования для буфера B0 (DMA_IF)

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
TDC_BLKCFG0_B0	Конфигурационный регистр 0	0x0	RW	0x0_0100
TDC_BLKCFG1_B0	Конфигурационный регистр 1	0x0	RW	0x0_0104
TDC_BLKCFG2_B0	Конфигурационный регистр 2	0x0	RW	0x0_0108
TDC_BLKCFG3_B0	Конфигурационный регистр 3	0x0	RW	0x0_010C
TDC_BLKCFG4_B0	Конфигурационный регистр 4	0x0	RW	0x0_0110
TDC_BLKCFG5_B0	Конфигурационный регистр 5	0x0	RW	0x0_0114
TDC_BLKCFG6_B0	Конфигурационный регистр 6	0x0	RW	0x0_0118
TDC_BLKCFG7_B0	Конфигурационный регистр 7	0x0	RW	0x0_011C
TDC_TRIG_B0	Регистр для запуска декодирования	0x0	WO	0x0_0120

Таблица 22.51. Регистры статуса и настроек декодирования для буфера B1 (DMA_IF)

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
TDC_BLKCFG0_B1	Конфигурационный регистр 0	0x0	RW	0x0_0200
TDC_BLKCFG1_B1	Конфигурационный регистр 1	0x0	RW	0x0_0204
TDC_BLKCFG2_B1	Конфигурационный регистр 2	0x0	RW	0x0_0208
TDC_BLKCFG3_B1	Конфигурационный регистр 3	0x0	RW	0x0_020C
TDC_BLKCFG4_B1	Конфигурационный регистр 4	0x0	RW	0x0_0210
TDC_BLKCFG5_B1	Конфигурационный регистр 5	0x0	RW	0x0_0214
TDC_BLKCFG6_B1	Конфигурационный регистр 6	0x0	RW	0x0_0218
TDC_BLKCFG7_B1	Конфигурационный регистр 7	0x0	RW	0x0_021C
TDC_TRIG_B1	Регистр для запуска декодирования	0x0	WO	0x0_0220

Таблица 22.52. Внутренняя память декодера (DMA_IF)

Условное обозначение	Описание	Тип доступа	Диапазон адресов
TDC_SYST_B0	Память систематических данных (X). Буфер B0	RW	0x1_0000 - 0x1_1FFF
TDC_PAR0_B0	Память проверочных данных 0 (A). Буфер B0	RW	0x1_2000 - 0x1_3FFF
TDC_PAR1_B0	Память проверочных данных 1 (B). Буфер B0	RW	0x1_4000 - 0x1_5FFF
TDC_PAR2_B0	Память проверочных данных 2 (C). Буфер B0	RW	0x1_6000 - 0x1_7FFF
TDC_PAR3_B0	Память проверочных данных 3 (A'). Буфер B0	RW	0x1_8000 - 0x1_9FFF
TDC_PAR4_B0	Память проверочных данных 4 (B'). Буфер B0	RW	0x1_A000 - 0x1_BFFF
TDC_PAR5_B0	Память проверочных данных 5 (C'). Буфер B0	RW	0x1_C000 - 0x1_DFFF
TDC_SYST_B1	Память систематических данных (X). Буфер B1	RW	0x2_0000 - 0x2_1FFF
TDC_PAR0_B1	Память проверочных данных 0 (A). Буфер B1	RW	0x2_2000 - 0x2_3FFF
TDC_PAR1_B1	Память проверочных данных 1 (B). Буфер B1	RW	0x2_4000 - 0x2_5FFF
TDC_PAR2_B1	Память проверочных данных 2 (C). Буфер B1	RW	0x2_6000 - 0x2_7FFF
TDC_PAR3_B1	Память проверочных данных 3 (A'). Буфер B1	RW	0x2_8000 - 0x2_9FFF
TDC_PAR4_B1	Память проверочных данных 4 (B'). Буфер B1	RW	0x2_A000 - 0x2_BFFF
TDC_PAR5_B1	Память проверочных данных 5 (C'). Буфер B1	RW	0x2_C000 - 0x2_DFFF
TDC_INTTBL_B0	Таблица перемежения. Буфер B0	RW	0x3_0000 - 0x3_3FFF
TDC_INTTBL_B1	Таблица перемежения. Буфер B1	RW	0x3_8000 - 0x3_BFFF
TDC_HDOUT_B0	Память выходных данных (жесткие решения). Буфер B0	RO	0x3_4000 - 0x3_43FF
TDC_HDOUT_B1	Память выходных данных (жесткие решения). Буфер B1	RO	0x3_C000 - 0x3_C3FF

22.4.4.2 Общие регистры статуса и управления

22.4.4.2.1 Регистр TDC_REVID

Таблица 22.53. Формат регистра TDC_REVID

Номер бита	Условное обозначение	Назначение
31:8	-	Резерв
7:4	X	Старшая часть номера версии (X)
3:0	Y	Младшая часть номера версии (Y)

22.4.4.2.2 Регистр TDC_SWRST

Таблица 22.54. Формат регистра TDC_SWRST

Номер бита	Условное обозначение	Назначение
31:1	-	Резерв
0	RESET	Запись 1 в этот бит инициирует программный сброс декодера. Выполняется инициализация автоматов и управляющих сигналов. Сброс не влияет на состояние программно-доступных RW регистров и памяти

22.4.4.2.3 Регистр TDC_STAT_B0/B1

Таблица 22.55. Формат регистра TDC_STAT_B0/B1

Номер бита	Условное обозначение	Назначение
31:2	-	Резерв
1	BUSY	Флаг занятости входного буфера
0	PEND	Флаг готовности соответствующего входного буфера к декодированию

22.4.4.2.4 Регистр TDC_EVSET_B0/B1

Таблица 22.56. Формат регистра TDC_EVSET_B0/B1

Номер бита	Условное обозначение	Назначение
31:1	-	Резерв
0	REVT	Запись 1 в этот бит устанавливает событие REVT для соответствующего буфера

22.4.4.2.5 Регистр TDC_IRQ_EN

Таблица 22.57. Формат регистра TDC_IRQ_EN

Номер бита	Условное обозначение	Назначение
31:2	-	Резерв
1	REVT_B1	Разрешение прерывания по событию REVT буфера 1
0	REVT_B0	Разрешение прерывания по событию REVT буфера 0

22.4.4.2.6 Регистр TDC_IRQ_STAT

Регистр содержит биты статуса для всех прерываний. Бит статуса устанавливается в 1 при возникновении прерывания без учета разрешений, либо при записи 1 в этот бит (эта возможность может быть использована для отладочных целей). Запись 0 игнорируется.

Таблица 22.58. Формат регистра TDC_IRQ_STAT

Номер бита	Условное обозначение	Назначение
31:2	-	Резерв
1	REVT_B1	Статус прерывания по событию REVT буфера 1
0	REVT_B0	Статус прерывания по событию REVT буфера 0

22.4.4.2.7 Регистр TDC_IRQ_EN_STAT

Регистр содержит биты статуса для всех разрешенных прерываний. Бит статуса устанавливается в 1 при возникновении прерывания только, если установлен соответствующий бит в регистре **TDC_IRQ_EN**. Регистр доступен только для чтения.

Таблица 22.59. Формат регистра TDC_IRQ_EN_STAT

Номер бита	Условное обозначение	Назначение
31:2	-	Резерв
1	REVT_B1	Статус прерывания по событию REVT буфера 1
0	REVT_B0	Статус прерывания по событию REVT буфера 0

22.4.4.2.8 Регистр TDC_IRQ_CLR

Регистр предназначен для программного сброса прерываний, бит статуса которых установлен в регистре **TDC_IRQ_STAT**. Сброс производится путем записи 1 в соответствующий бит. Запись 0 игнорируется. При чтении регистра возвращается 0.

Таблица 22.60. Формат регистра TDC_IRQ_CLR

Номер бита	Условное обозначение	Назначение
31:2	-	Резерв
1	REVT_B1	Сброс прерывания по событию REVT буфера 1
0	REVT_B0	Сброс прерывания по событию REVT буфера 0

22.4.4.3 Регистры настроек декодирования

22.4.4.3.1 Регистр TDC_BLKCFG0

Таблица 22.61. Формат регистра TDC_BLKCFG0

Номер бита	Условное обозначение	Назначение
31:13	-	Резерв
12:0	BLK_LEN	Размер блока минус 1. Допустимый диапазон значений: 39 - 8191

22.4.4.3.2 Регистр TDC_BLKCFG1

Таблица 22.62. Формат регистра TDC_BLKCFG1

Номер бита	Условное обозначение	Назначение
31:22	-	Резерв
21:20	SW2_LEN_SEL	Размер окна SW2: 0: окно SW2 не используется, 1: SW2_LEN = SW1_LEN, 2: SW2_LEN = SW1_LEN - 2;

		3: резерв
19	-	Резерв
18:12	SW1_LEN	Размер окна SW1: 0 - 8: недопустимые значения; 9 - 127: 10 - 128 бит
11	-	Резерв
10:8	SW0_LEN_SEL	Размер скользящего окна SW0 0: 16 бит, 1: 32 бита, 2: 48 бит, 3: 64 бита, 4: 96 бит, 5: 128 бит, 6, 7: резерв
7:6	-	Резерв
5:0	NUM_SW0	Количество окон SW0: 0 - 62. 0 - окно SW0 не используется. Значение 63 не допустимо

22.4.4.3.3 Регистр TDC_BLKCFG2

Таблица 22.63. Формат регистра TDC_BLKCFG2

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:14	RATE_SEL	Выбор скорости кодирования: 0: 1/3; 1: 1/5; 2: 1/7; 3: резерв
13:12	ITG_MODE	Режим генерации таблицы перемежения: 0: используется таблица, загруженная во внутреннюю память декодера. Закон перемежения произвольный; 1: используется таблица, загруженная во внутреннюю память декодера. Закон перемежения сохраняет четность индексов; 2: используется внутренний генератор; 3: резерв
11	INTRL_LOAD	Управление загрузкой таблицы перемежения: 0: таблица не загружается, для перемежения используются ранее записанные индексы; 1: таблица перемежения загружается с помощью DMA
10:4	-	Резерв
3:0	NUM_ITER	Количество итераций минус 1

22.4.4.3.4 Регистр TDC_BLKCFG3

Таблица 22.64. Формат регистра TDC_BLKCFG3

Номер бита	Условное обозначение	Назначение
31:24	BETA_ST3_MAP0	Начальные значения beta для состояний 0 - 3 первой половины итерации
23:16	BETA_ST2_MAP0	
15:8	BETA_ST1_MAP0	
7:0	BETA_ST0_MAP0	

22.4.4.3.5 Регистр TDC_BLKCFG4

Таблица 22.65. Формат регистра TDC_BLKCFG4

Номер бита	Условное обозначение	Назначение

31:24	BETA_ST7_MAP0	Начальные значения beta для состояний 4 - 7 первой половины итерации
23:16	BETA_ST6_MAP0	
15:8	BETA_ST5_MAP0	
7:0	BETA_ST4_MAP0	

22.4.4.3.6 Регистр TDC_BLKCFG5

Таблица 22.66. Формат регистра TDC_BLKCFG5

Номер бита	Условное обозначение	Назначение
31:24	BETA_ST3_MAP1	Начальные значения beta для состояний 0 - 3 второй половины итерации
23:16	BETA_ST2_MAP1	
15:8	BETA_ST1_MAP1	
7:0	BETA_ST0_MAP1	

22.4.4.3.7 Регистр TDC_BLKCFG6

Таблица 22.67. Формат регистра TDC_BLKCFG6

Номер бита	Условное обозначение	Назначение
31:24	BETA_ST7_MAP1	Начальные значения beta для состояний 4 - 7 второй половины итерации
23:16	BETA_ST6_MAP1	
15:8	BETA_ST5_MAP1	
7:0	BETA_ST4_MAP1	

22.4.4.3.8 Регистр TDC_BLKCFG7

Таблица 22.68. Формат регистра TDC_BLKCFG7

Номер бита	Условное обозначение	Назначение
31:29	-	Резерв
28:16	ITG_P1	Параметр генератора таблицы перемежения по стандарту LTE: $(f_1 + f_2) \bmod K$
15:13	-	Резерв
12:0	ITG_P0	Параметр генератора таблицы перемежения по стандарту LTE: $(2 * f_2) \bmod K$

22.4.4.4 Регистры запуска декодирования

22.4.4.4.1 Регистр TDC_TRIG

Таблица 22.69. Формат регистра TDC_TRIG

Номер бита	Условное обозначение	Назначение
31:1	-	Резерв
0	TRIG	Запись 1 в этот бит инициирует процесс декодирования

22.5 Турбо кодер (TEC)

22.5.1 Общая информация

22.5.1.1 Основные особенности

Блок TEC (Turbo Encoder Coprocessor) предназначен для выполнения помехоустойчивого кодирования данных с использованием турбо-кода.

TEC имеет следующие характеристики:

- Поддержка турбо кодов со скоростью 1/3, 1/5, 1/7
- Поддерживаемое кодовое ограничение: 4
- Размер кодируемого блока данных: 40 – 8192 бит
- Поддержка перемежения по конфигурируемой таблице
- Внутренний квадратурный перемежитель
- Два набора входных/выходных буферов, а также ядер кодера для увеличения производительности
- Аппаратный интерфейс для работы с DMA
- Производительность: не менее 100 Мбит/с

22.5.1.2 Структура кода

Структура турбо кодера показана на Рисунок 22.26. Турбо кодер состоит из двух рекурсивных систематических сверточных (Recursive Systematic Convolutional – RSC) кодеров, соединенных параллельно, и перемежителя. Каждый составной кодер имеет кодовое ограничение 4 и четыре выхода: систематические данные (X_0, X_1) и паритетные данные (A_0, A_1, B_0, B_1, C_0, C_1).

На вход первого RSC кодера поступает последовательность бит непосредственно со входа X. На вход второго RSC кодера данные поступают с выхода перемежителя (Interleaver), переставляющего биты исходного сообщения по заданному закону.

На выходе турбо кодера выполняется перфорация данных для получения необходимой кодовой скорости. Возможны следующие варианты выходной последовательности бит:

- кодовая скорость 1/3: $X_0[0]$, $A_0[0]$, $A_1[0]$, $X_0[1]$, $A_0[1]$, $A_1[1]$, ...
- кодовая скорость 1/5: $X_0[0]$, $A_0[0]$, $B_0[0]$, $A_1[0]$, $B_1[0]$, $X_0[1]$, $A_0[1]$, $B_0[1]$, $A_1[1]$, $B_1[1]$, ...
- кодовая скорость 1/7: $X_0[0]$, $A_0[0]$, $B_0[0]$, $C_0[0]$, $A_1[0]$, $B_1[0]$, $C_1[0]$, $X_0[1]$, $A_0[1]$, $B_0[1]$, $C_0[1]$, $A_1[1]$, $B_1[1]$, $C_1[1]$, ...

Перед кодированием блока данных оба составных кодера находятся в нулевом состоянии. После кодирования блока оба кодера также переводятся в нулевое состояние. Это осуществляется с помощью переключателей, показанных на Рисунок 22.26. В течение первых трех тактов переключатель первого RSC кодера находится в нижнем положении, и на выход передаются биты X_0, A_0, B_0, C_0 («хвостовые» биты). В течение следующих трех тактов переключатель второго RSC кодера находится в нижнем положении, и на выход передаются биты X_1, A_1, B_1, C_1.

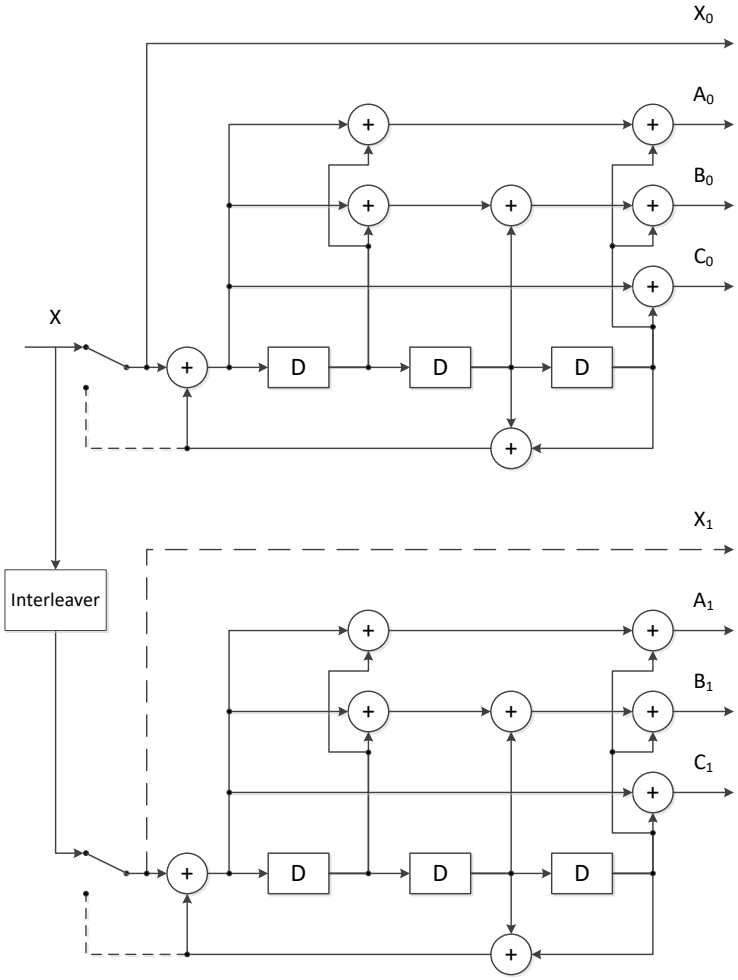


Рисунок 22.26. Структура турбо кодера

22.5.2 Структурная схема

На Рисунок 22.27 показана структурная схема TEC.

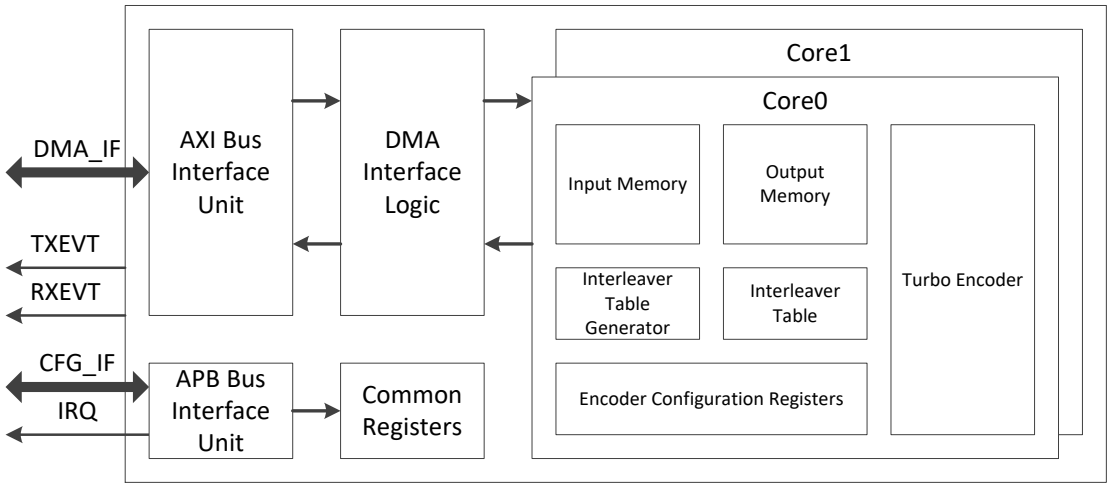


Рисунок 22.27. Структурная схема TEC

Основные составляющие блоки TEC и их функции:

- AXI Bus Interface Unit: блок реализует slave-устройство на интерфейсе данных DMA_IF.
- APB Bus Interface Unit: блок реализует slave-устройство на конфигурационном интерфейсе CFG_IF.
- DMA Interface Logic: реализует управление загрузкой, выгрузкой данных и конфигурации по интерфейсу DMA_IF.
- Common Registers: общие регистры статуса, программного сброса и прерываний.
- Input Memory: память для хранения входных данных кодера.
- Output Memory: память для хранения выходных данных кодера.
- Interleaver Table Generator: внутренний генератор таблицы перемежения.
- Interleaver Table: память для хранения таблицы перемежения.
- Encoder Configuration Registers: регистры параметров кодирования.
- Turbo Encoder: блок турбо-кодирования.

22.5.3 Функционирование

22.5.3.1 Введение

Типовой сценарий использования блока TEC предполагает работу совместно с системным контроллером DMA посредством сигналов событий TXEVT и RXEVT, которые соответствуют двум направлениям передачи данных: TX – из системной памяти во внутреннюю память TEC (параметры кодирования и входные данные), RX – из памяти TEC в системную память (выходных данные).

Общий алгоритм использования блока TEC состоит из следующих шагов:

1. Хост-процессор (CPU) выполняет инициализацию кодера через интерфейс CFG_IF: включение рабочего режима (MODE = 1), разрешение прерываний, при необходимости.
2. CPU настраивает системный блок DMA для передачи в TEC параметров кодирования, таблицы перемежения и данных кодируемого блока (или набора из BLK_CNT + 1 блоков), а также для чтения кодированных блоков из TEC.
3. После подготовки входных данных для кодирования CPU запускает контроллер DMA для передачи конфигурационных параметров кодируемых блоков и таблицы перемежения (если используется внешняя таблица).
4. TEC сигнализирует контроллеру DMA о готовности к приему первого кодируемого блока данных с помощью события TXEVT. После получения данных TEC начинает кодирование блока.
5. Если второе ядро кодера в данный момент не занято, и еще имеются блоки для кодирования (зависит от BLK_CNT), то TEC генерирует еще одно событие TXEVT для передачи данных следующего блока.
6. По завершению кодирования первого блока TEC генерирует событие RXEVT, что запускает контроллер DMA на чтение кодированных данных из внутренней памяти TEC. Если после этого еще имеются блоки для кодирования (зависит от BLK_CNT), выполняется переход к шагу 4.
7. После обработки и чтения всех (BLK_CNT + 1) блоков контроллер DMA генерирует прерывание в CPU.
8. Если имеется очередной набор блоков для кодирования, то выполняется переход к шагу 2.

Далее приводится более подробное описание этих шагов.

22.5.3.2 Работа с DMA

Настройка контроллера DMA заключается в подготовке дескрипторов задач для пересылки необходимой информации.

В направлении TX выполняются следующие пересылки:

- конфигурационные параметры, записываемые в регистры **TEC_BLKCFG***. Необходимо записывать все регистры, даже если их значения не изменялись с предыдущего запуска кодера;
- таблица перемежения (см. пункт 22.5.3.6.2);
- один или несколько блоков входных данных.

В направлении RX выполняется пересылка одного или нескольких блоков выходных данных.

На Рисунок 22.28 показан пример последовательности передачи данных при кодировании трех блоков (BLK_CNT = 2) с идентичным набором параметров. Здесь CFG – конфигурационные параметры, TBL – таблица перемежения, DI #i – блоки входных данных, DO #i –блоки выходных данных.

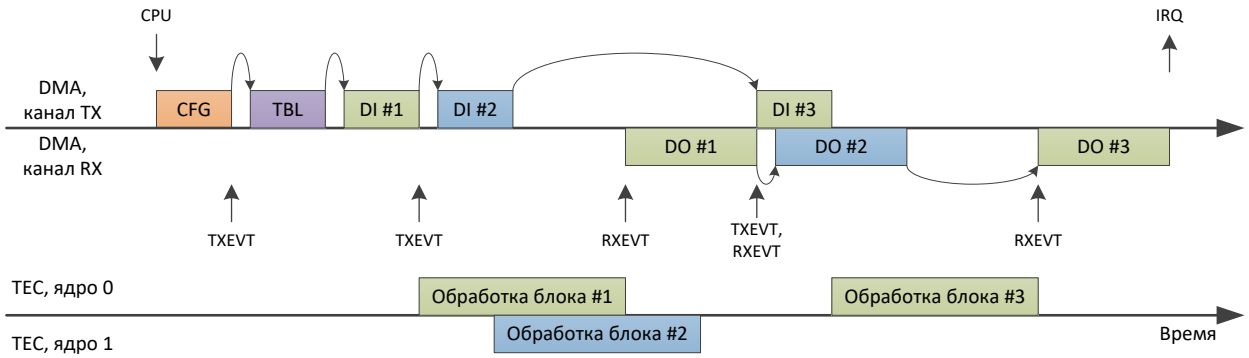


Рисунок 22.28. Пример кодирования блоков с одиннаковыми параметрами

На Рисунок 22.29 показан пример последовательности передачи данных при кодировании нескольких блоков с различными параметрами. При этом для кодирования блока #1 используется внешняя таблица перемежения, а для кодирования блока #2 – внутренняя.

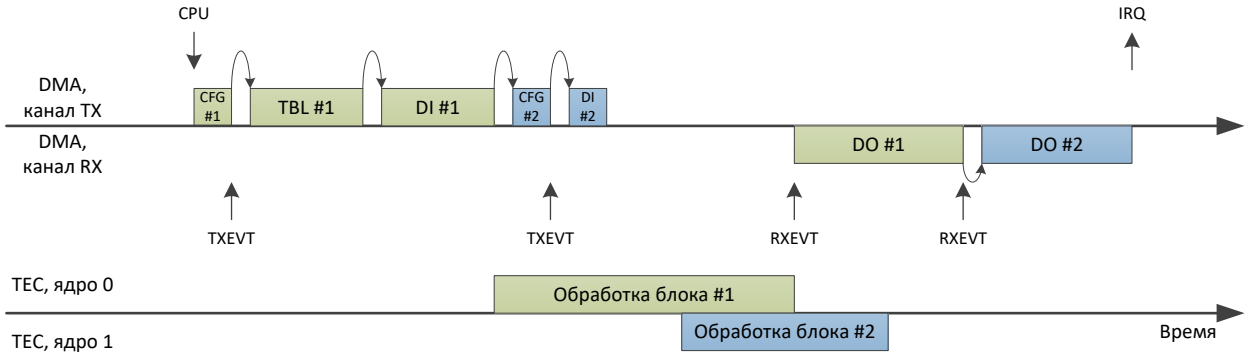


Рисунок 22.29. Пример кодирования блоков с различными параметрами

22.5.3.3 Двухбуферная схема

Блок TEC содержит два ядра турбо-кодеров: Core 0, Core 1. Каждое ядро имеет собственный набор конфигурационных регистров, память входных и выходных данных (Рисунок 22.30).

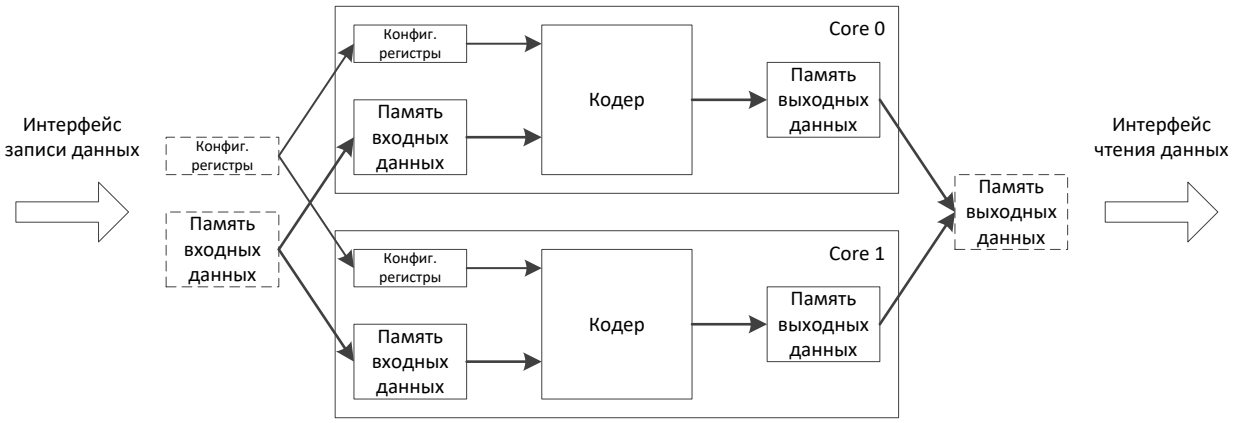


Рисунок 22.30. Функциональная схема TEC

В рабочем режиме (параметр MODE = 1, регистр **TEC_CFG**) выбор ядра производится внутренней логикой TEC по принципу чередования ядер для каждого нового блока входных данных. При этом доступ к регистрам и памяти активного ядра осуществляется по адресам ядра 0. Для целей отладки предусмотрен режим непосредственного доступа к памяти каждого ядра (MODE = 0).

22.5.3.4 Формат входных/выходных данных

Блок TEC поддерживает несколько форматов упаковки данных в системной памяти. Преобразование формата выполняется автоматически при загрузке и выгрузке данных. За формат входных и выходных данных отвечают параметры IN_ORDER и OUT_ORDER регистра **TEC_BLKCFG0** соответственно.

На рис Рисунок 22.31 показаны варианты расположения бит данных в системной памяти в зависимости от значений параметров IN_ORDER и OUT_ORDER. Бит 0 является первым битом, передаваемым в блоке данных.

IN_/OUT_ORDER = 0
(исходный порядок)

msb							lsb	
127	120	119	112	111	104	103	96	Base + 0xC
95	88	87	80	79	72	71	64	Base + 0x8
63	56	55	48	47	40	39	32	Base + 0x4
31	24	23	16	15	8	7	0	Base + 0x0

IN_/OUT_ORDER = 2
(обратный порядок байт в слове)

msb							lsb	
103	96	111	104	119	112	127	120	Base + 0xC
71	64	79	72	87	80	95	88	Base + 0x8
39	32	47	40	55	48	63	56	Base + 0x4
7	0	15	8	23	16	31	24	Base + 0x0

IN_/OUT_ORDER = 1
(обратный порядок бит в байте)

msb							lsb	
120	127	112	119	104	111	96	103	Base + 0xC
88	95	80	87	72	79	64	71	Base + 0x8
56	63	48	55	40	47	32	39	Base + 0x4
24	31	16	23	8	15	0	7	Base + 0x0

IN_/OUT_ORDER = 3
(обратный порядок бит и байт)

msb							lsb	
96	103	104	111	112	119	120	127	Base + 0xC
64	71	72	79	80	87	88	95	Base + 0x8
32	39	40	47	48	55	56	63	Base + 0x4
0	7	8	15	16	23	24	31	Base + 0x0

Рисунок 22.31. Формат входных/выходных данных

22.5.3.5 Порядок выходных данных в памяти кодера

В таблицах ниже показан порядок расположения кодированных данных в выходной памяти TEC в зависимости от скорости кодирования. Здесь K – размер блока, биты K, K + 1, K + 2 соответствуют теил битам.

Таблица 22.70. Расположение кодированных данных в памяти для скорости 1/3

Индекс бита	SYST	PAR0	PAR1	PAR2	PAR3	PAR4	PAR5
0	$X_0[0]$	$A_0[0]$	-	-	$A_1[0]$	-	-
1	$X_0[1]$	$A_0[1]$	-	-	$A_1[1]$	-	-
…	…	…	…	…	…	…	…
K – 1	$X_0[K - 1]$	$A_0[K - 1]$	-	-	$A_1[K - 1]$	-	-
K	$X_0[K]$	$A_0[K]$	-	-	$X_0[K + 1]$	-	-
K + 1	$A_0[K + 1]$	$X_0[K + 2]$	-	-	$A_0[K + 2]$	-	-
K + 2	$X_1[K]$	$A_1[K]$	-	-	$X_1[K + 1]$	-	-
K + 3	$A_1[K + 1]$	$X_1[K + 2]$	-	-	$A_1[K + 2]$	-	-

Таблица 22.71. Расположение кодированных данных в памяти для скорости 1/5

Индекс бита	SYST	PAR0	PAR1	PAR2	PAR3	PAR4	PAR5
0	$X_0[0]$	$A_0[0]$	$B_0[0]$	-	$A_1[0]$	$B_1[0]$	-
1	$X_0[1]$	$A_0[1]$	$B_0[1]$	-	$A_1[1]$	$B_1[1]$	-
…	…	…	…	…	…	…	…
K – 1	$X_0[K - 1]$	$A_0[K - 1]$	$B_0[K - 1]$	-	$A_1[K - 1]$	$B_1[K - 1]$	-
K	$X_0[K]$	$A_0[K]$	$B_0[K]$	-	$X_0[K + 1]$	$A_0[K + 1]$	-
K + 1	$B_0[K + 1]$	$X_0[K + 2]$	$A_0[K + 2]$	-	$B_0[K + 2]$	$X_1[K]$	-
K + 2	$A_1[K]$	$B_1[K]$	$X_1[K + 1]$	-	$A_1[K + 1]$	$B_1[K + 1]$	-
K + 3	$X_1[K + 2]$	$A_1[K + 2]$	$B_1[K + 2]$	-	-	-	-

Таблица 22.72. Расположение кодированных данных в памяти для скорости 1/7

Индекс бита	SYST	PAR0	PAR1	PAR2	PAR3	PAR4	PAR5
0	$X_0[0]$	$A_0[0]$	$B_0[0]$	$C_0[0]$	$A_1[0]$	$B_1[0]$	$C_1[0]$
1	$X_0[1]$	$A_0[1]$	$B_0[1]$	$C_0[1]$	$A_1[1]$	$B_1[1]$	$C_1[1]$
…	…	…	…	…	…	…	…

Индекс бита	SYST	PAR0	PAR1	PAR2	PAR3	PAR4	PAR5
K – 1	$X_0[K - 1]$	$A_0[K - 1]$	$B_0[K - 1]$	$C_0[K - 1]$	$A_1[K - 1]$	$B_1[K - 1]$	$C_1[K - 1]$
K	$X_0[K]$	$A_0[K]$	$B_0[K]$	$C_0[K]$	$X_0[K + 1]$	$A_0[K + 1]$	$B_0[K + 1]$
K + 1	$C_0[K + 1]$	$X_0[K + 2]$	$A_0[K + 2]$	$B_0[K + 2]$	$C_0[K + 2]$	$X_1[K]$	$A_1[K]$
K + 2	$B_1[K]$	$C_1[K]$	$X_1[K + 1]$	$A_1[K + 1]$	$B_1[K + 1]$	$C_1[K + 1]$	$X_1[K + 2]$
K + 3	$A_1[K + 2]$	$B_1[K + 2]$	$C_1[K + 2]$	-	-	-	-

22.5.3.6 Таблица перемежения

Таблица перемежения может генерироваться внутри TEC либо храниться во внешней (по отношению к TEC) памяти и загружаться с помощью блока DMA.

22.5.3.6.1 Внутренняя генерация таблицы перемежения

TEC имеет встроенный генератор таблицы перемежения, включаемый при установке параметра ITG_MODE = 1 (регистр **TEC_BLKCFG0**). Закон перемежения соответствует стандарту LTE (ETSI TS 136 212 V10.0.0) и задается следующей формулой:

$$p(i) = (f_1 * i + f_2 * i^2) \bmod K,$$

где f_1, f_2 – параметры, определяемые по Таблица 22.48 в зависимости от размера блока; K – размер кодового блока; i – порядковый индекс до перемежения; p(i) – порядковый индекс после перемежения; mod – операция деления по модулю.

Для работы генератора предварительно необходимо рассчитать несколько параметров, перечисленных в Таблица 22.47.

Таблица 22.73. Параметры внутреннего генератора таблицы перемежения блока TEC

Регистр	Параметр	Значение
INCFG1	ITG_P0	$(2 * f_2) \bmod K$
INCFG1	ITG_P1	$(f_1 + f_2) \bmod K$

22.5.3.6.2 Внешняя генерация таблицы перемежения

Таблица перемежения также может генерироваться с помощью DSP или храниться во внешней памяти, а затем загружаться в TEC с помощью блока DMA. Таблица представляет собой массив 13-ти битных целочисленных индексов. В памяти TEC каждый индекс занимает два байта, при этом старшие 3 бита не используются.

При использовании внешней таблицы в параметр ITG_MODE необходимо записать значение 0.

Для каждого набора из (BLK_CNT + 1) блоков может использоваться и загружаться своя таблица перемежения (TBL_LOAD = 1). Если же для всех блоков используется только одна общая таблица, то ее можно загрузить один раз во время инициализации кодера (TBL_LOAD = 0).

22.5.3.7 Программный сброс

При необходимости, программный сброс выполняется путем записи значения 1 в регистр **TEC_SWRST**. Программный сброс приводит TEC в начальное состояние, но не влияет на состояние памяти и регистров **TEC_BLKCFG**.

Для использования прерываний необходимо установить соответствующие разряды в регистре **TEC_IRQ_EN**.

22.5.3.8 Прерывания

Аппаратные сигналы TXEVT, RXEVT также могут генерировать прерывание. Эту возможность можно использовать для отладки или для альтернативного (более медленного) способа работы с TEC без участия контроллера DMA.

Для использования прерываний необходимо установить соответствующие разряды в регистре **TEC_IRQ_EN**. Состояние разрешенных прерываний отображается в регистре **TEC_IRQ_EN_STAT**. Для сброса активного прерывания необходимо записать '1' в соответствующий разряд регистра **TEC_IRQ_CLR**.

22.5.3.9 Средства отладки

Для целей отладки доступны следующие средства:

- при выключенном кодере (MODE = 0) имеется программный доступ к памяти обоих ядер по соответствующим адресам;
- принудительная генерация событий TXEVT, RXEVT с помощью регистров **TEC_TXES**, **TEC_RXES**;
- отладочная информация в регистре **TEC_STAT**.

22.5.4 Регистры

22.5.4.1 Перечень регистров и блоков памяти

В Таблица 22.74 перечислены общие регистры статуса и управления TEC. Доступ к регистрам возможен через интерфейс CFG_IF только 32-х разрядными словами.

Таблица 22.74. Регистры статуса и управления TEC (интерфейс CFG_IF)

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
TEC_REVID	Версия RTL	0x0[1]	RO	0x0000
TEC_SWRST	Программный сброс	0x0	WO	0x0004
TEC_CFG	Регистр общих настроек кодера	0x0	RW	0x0008
TEC_STAT	Регистр статуса кодера	0x0	RO	0x000C
TEC_TXES	Регистр установки события TXEVT	0x0	WO	0x0010
TEC_RXES	Регистр установки события RXEVT	0x0	WO	0x0014
TEC_IRQ_EN	Регистр разрешения прерываний	0x0	RW	0x0018
TEC_IRQ_STAT	Регистр статуса/установки прерываний	0x0	RW	0x001C
TEC_IRQ_EN_STAT	Регистр статуса разрешенных прерываний	0x0	RO	0x0020
TEC_IRQ_CLR	Регистр сброса прерываний	0x0	WO	0x0024

Примечания:
1) Версия RTL имеет формат X.Y и соответствует версии SVN релиза. Например, для релиза r1p2 X = 1, Y = 2

В Таблица 22.75, Таблица 22.76 перечислены регистры для настройки параметров кодирования и блоки памяти TEC. Доступ к регистрам и памяти производится по интерфейсу DMA_IF..

Доступ к регистрам **TEC_BLKCFG** возможен только 32-х, 64-х и 128-ми разрядными словами. В рабочем режиме (MODE = 1) TEC поддерживает только 128-ми разрядные обращения к памяти. При MODE = 0 обращения к памяти могут быть с разрядностью 8, 16, 32, 64, 128 бит.

Запись по неиспользуемым адресам игнорируется.

Таблица 22.75. Регистры конфигурации кодового блока TEC (интерфейс DMA_IF)

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
TEC_BLKCFG0	Конфигурационный регистр 0	0x0	RW	0x0000
TEC_BLKCFG1	Конфигурационный регистр 1	0x0	RW	0x0004

Таблица 22.76. Внутренняя память TEC (интерфейс DMA_IF)

Условное обозначение	Описание	Тип доступа	Диапазон адресов
Отладочный режим: доступ к памяти ядра 0 Рабочий режим: доступ к обобщенной памяти TEC, выбор ядра выполняется внутренней логикой			
TEC_INMEM_B0	Память входных данных	RW	0x1000 - 0x13FF
TEC_INTTBL_B0	Таблица перемежения	RW	0x2000 - 0x5FFF

Условное обозначение	Описание	Тип доступа	Диапазон адресов
TEC_SYST_B0	Память систематических данных (X_0)	RW	0xA000 - 0xA400
TEC_PAR0_B0	Память проверочных данных 0 (A_0)	RW	0xA500 - 0xA900
TEC_PAR1_B0	Память проверочных данных 1 (B_0)	RW	0xAA00 - 0xAE00
TEC_PAR2_B0	Память проверочных данных 2 (C_0)	RW	0xAF00 - 0xB300
TEC_PAR3_B0	Память проверочных данных 3 (A_1)	RW	0xB400 - 0xB800
TEC_PAR4_B0	Память проверочных данных 4 (B_1)	RW	0xB900 - 0xBD00
TEC_PAR5_B0	Память проверочных данных 5 (C_1)	RW	0xBE00 - 0xC200
Отладочный режим: доступ к памяти ядра 1 Рабочий режим: адреса не доступны			
TEC_INMEM_B1	Память входных данных	RW	0x1400 - 0x17FF
TEC_INTTBL_B1	Таблица перемежения	RW	0x6000 - 0x9FFF
TEC_SYST_B1	Память систематических данных (X_0)	RW	0xC300 - 0xC700
TEC_PAR0_B1	Память проверочных данных 0 (A_0)	RW	0xC800 - 0xCC00
TEC_PAR1_B1	Память проверочных данных 1 (B_0)	RW	0xCD00 - 0xD100
TEC_PAR2_B1	Память проверочных данных 2 (C_0)	RW	0xD200 - 0xD600
TEC_PAR3_B1	Память проверочных данных 3 (A_1)	RW	0xD700 - 0xDB00
TEC_PAR4_B1	Память проверочных данных 4 (B_1)	RW	0xDC00 - 0xE000
TEC_PAR5_B1	Память проверочных данных 5 (C_1)	RW	0xE100 - 0xE500

22.5.4.2 Регистры статуса и управления

22.5.4.2.1 Регистр TEC_REVID

Таблица 22.77. Формат регистра TEC_REVID

Номер бита	Условное обозначение	Назначение
31:8	-	Резерв
7:4	X	Старшая часть номера версии (X)
3:0	Y	Младшая часть номера версии (Y)

22.5.4.2.2 Регистр TEC_SWRST

Таблица 22.78. Формат регистра TEC_SWRST

Номер бита	Условное обозначение	Назначение
31:1	-	Резерв
0	SOFT_RESET	Запись 1 в этот бит инициирует программный сброс блока. Сброс не влияет на состояние регистров **TEC_INCFG** и памяти

22.5.4.2.3 Регистр TEC_CFG

Таблица 22.79. Формат регистра TEC_CFG

Номер бита	Условное обозначение	Назначение
31:3	-	Резерв
2	TBL_LOAD	Управление загрузкой таблицы перемежения: 0: таблица перемежения загружается один раз перед включением кодера; 1: таблица перемежения загружается для каждого набора из BLK_CNT блоков
1	-	Резерв
0	MODE	Режим функционирования кодера: 0: кодер выключен, возможен доступ к памяти всех ядер; 1: рабочий режим

22.5.4.2.4 Регистр TEC_STAT

Таблица 22.80. Формат регистра TEC_STAT

Номер бита	Условное обозначение	Назначение
31:6	-	Резерв
5	OBUF_1_ACTIVE	Состояние выходного буфера 1: 0: буфер не активен; 1: буфер находится в ожидании чтения выходных данных (было сгенерировано событие RXEVT)
4	ENC_1_ACTIVE	Состояние кодера 1: 0: кодер не активен; 1: кодер активен (выполняется кодирование данных)
3	IBUF_1_ACTIVE	Состояние входного буфера 1: 0: буфер не активен; 1: буфер находится в ожидании записи входных данных (было сгенерировано событие TXEVT)
2	OBUF_0_ACTIVE	Состояние выходного буфера 0: 0: буфер не активен; 1: буфер находится в ожидании чтения выходных данных (было сгенерировано событие RXEVT)
1	ENC_0_ACTIVE	Состояние кодера 0: 0: кодер не активен; 1: кодер активен (выполняется кодирование данных)
0	IBUF_0_ACTIVE	Состояние входного буфера 0: 0: буфер не активен; 1: буфер находится в ожидании записи входных данных (было сгенерировано событие TXEVT)

22.5.4.2.5 Регистр TEC_TXES

Таблица 22.81. Формат регистра TEC_TXES

Номер бита	Условное обозначение	Назначение
31:1	-	Резерв
0	TXEVT	Запись 1 в этот бит генерирует событие TXEVT

22.5.4.2.6 Регистр TEC_RXES

Таблица 22.82. Формат регистра TEC_RXES

Номер бита	Условное обозначение	Назначение
31:1	-	Резерв
0	RXEVT	Запись 1 в этот бит генерирует событие RXEVT
0	TXEVT	

22.5.4.2.7 Регистр TEC_IRQ_EN

Таблица 22.83. Формат регистра TEC_IRQ_EN

Номер бита	Условное обозначение	Назначение
31:2	-	Резерв
1	RXEVT	Разрешение прерывания по событию RXEVT
0	TXEVT	Разрешение прерывания по событию TXEVT

22.5.4.2.8 Регистр TEC_IRQ_STAT

Регистр содержит биты статуса для всех прерываний. Бит статуса устанавливается в 1 при возникновении прерывания без учета разрешений, либо при записи 1 в этот бит (эта возможность может быть использована для отладочных целей). Запись 0 игнорируется.

Таблица 22.84. Формат регистра TEC_IRQ_STAT

Номер бита	Условное обозначение	Назначение
31:2	-	Резерв
1	RXEVT	Статус прерывания по событию RXEVT
0	TXEVT	Статус прерывания по событию TXEVT

22.5.4.2.9 Регистр TEC_IRQ_EN_STAT

Регистр содержит биты статуса для всех разрешенных прерываний. Бит статуса устанавливается в 1 при возникновении прерывания только, если установлен соответствующий бит в регистре **TEC_IRQ_EN**. Регистр доступен только для чтения.

Таблица 22.85. Формат регистра TEC_IRQ_EN_STAT

Номер бита	Условное обозначение	Назначение
31:2	-	Резерв
1	RXEVT	Статус прерывания по событию RXEVT
0	TXEVT	Статус прерывания по событию TXEVT

22.5.4.2.10 Регистр TEC_IRQ_CLR

Регистр предназначен для программного сброса прерываний, бит статуса которых установлен в регистре **TEC_IRQ_STAT**. Сброс производится путем записи 1 в соответствующий бит. Запись 0 игнорируется. При чтении регистра возвращается 0.

Таблица 22.86. Формат регистра CENC_IRQ_CLR

Номер бита	Условное обозначение	Назначение
31:2	-	Резерв
1	RXEVT	Сброс прерывания по событию RXEVT
0	TXEVT	Сброс прерывания по событию TXEVT

22.5.4.3 Регистры настроек кодирования

22.5.4.3.1 Регистр TEC_BLKCFG0

Таблица 22.87. Формат регистра TEC_BLKFG0

Номер бита	Условное обозначение	Назначение
31	-	Резерв
30:29	OUT_ORDER	Порядок бит выходных данных: 0: исходный порядок бит; 1: обратный порядок бит в каждом байте; 2: исходный порядок бит, обратный порядок байт в каждом слове; 3: обратный порядок бит в байте, обратный порядок байт в слове
28:27	IN_ORDER	Порядок бит входных данных: 0: исходный порядок бит;

		1: обратный порядок бит в каждом байте; 2: исходный порядок бит, обратный порядок байт в каждом слове; 3: обратный порядок бит в байте, обратный порядок байт в слове
26	ITG_MODE	Режим генерации таблицы перемежения: 0: используется внешняя таблица, загруженная в память TEC; 1: используется внутренний генератор таблицы перемежения
25:24	RATE_SEL	Выбор скорости кодирования: 0: 1/3; 1: 1/5; 2: 1/7; 3: резерв
23:16	BLK_CNT	Количество блоков в наборе (значение минус 1)
15:13	-	Резерв
12:0	BLK_LEN	Размер блока (значение минус 1)

22.5.4.3.2 Регистр TEC_BLKCFG1

Таблица 22.88. Формат регистра TEC_BLKFG1

Номер бита	Условное обозначение	Назначение
31:29	-	Резерв
28:16	ITG_P1	Параметр генератора таблицы перемежения по стандарту LTE: $(f_1 + f_2) \bmod K$
15:13	-	Резерв
12:0	ITG_P0	Параметр генератора таблицы перемежения по стандарту LTE: $(2 * f_2) \bmod K$

22.6 Быстрое преобразование Фурье (FFT)

22.6.1 Общая информация

Назначением блока FFT является автономное, параллельное с работой DSP-процессора выполнение быстрых преобразований Фурье (БПФ) комплексных массивов, а также некоторых сопутствующих операций.

Блок FFT имеет следующие особенности:

- ввод/вывод выполняются в реальном времени, параллельно с обработкой;
- входные/выходные данные для пользователя располагаются в прямом порядке;
- поддерживаются следующие форматы действительных/мнимых компонент входных и выходных данных:
  - 32-разрядная плавающая точка (стандарт IEEE-754);
  - 32-разрядное целое число (дополнительный код);
  - 16-разрядное целое число (дополнительный код);
- формат вычислений: 32-разрядная плавающая точка.
- максимальный размер непосредственно выполняемого преобразования – 8192;
- минимальный размер преобразования – 16;
- максимальный размер наращиваемого преобразования – 256K;
- двухбуферная схема работы.

Блок FFT имеет возможность выполнять следующие сопутствующие операции:

- обратное быстрое преобразование Фурье;
- нормировка результатов преобразования;
- фазовые матричные повороты результатов преобразования;
- расчет мощностей результатов преобразования;
- одновременное преобразование $M = 2^m$ комплексных массивов;
- поэлементное перемножение (с сопряжением) двух комплексных массивов;
- быстрая БПФ – свертка (ковариация);
- вычисление скалярного произведения комплексных массивов.

22.6.2 Структурная схема

На Рисунок 22.32 показана структурная схема FFT.

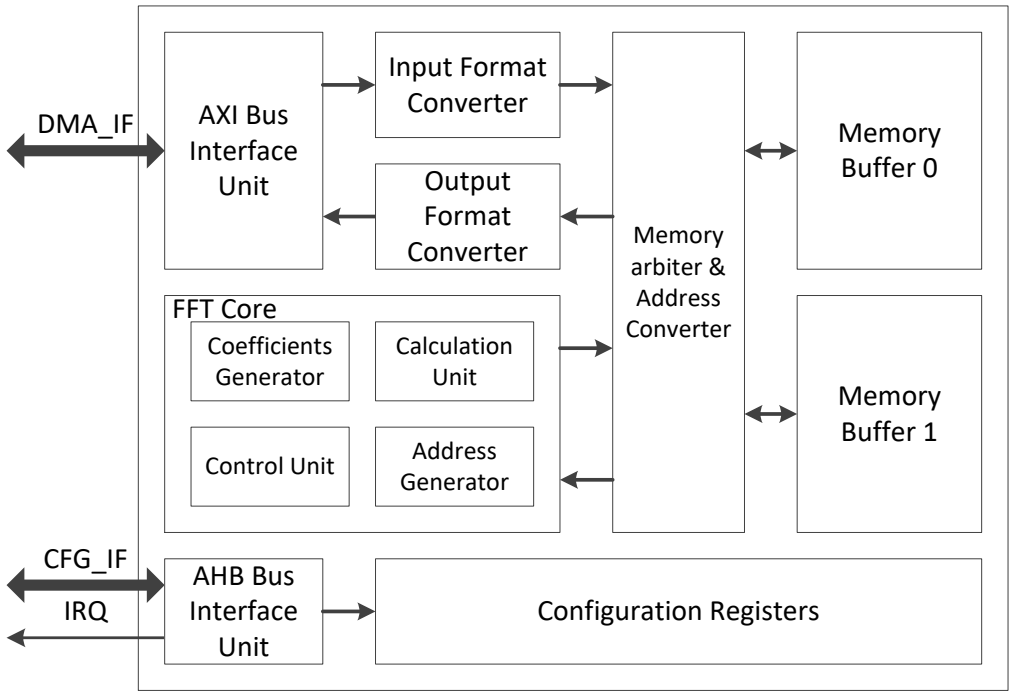


Рисунок 22.32. Структурная схема FFT

Основные составляющие блоки FFT и их функции:

- AXI Bus Interface Unit: блок реализует slave-устройство на интерфейсе данных DMA_IF.
- AHB Bus Interface Unit: блок реализует slave-устройство на конфигурационном интерфейсе CFG_IF.
- Configuration Registers: конфигурационные регистры.
- Input Format Converter: выполняет преобразование формата входных данных во внутренний формат вычислений.
- Output Format Converter: выполняет преобразование внутреннего формата вычислений в формат выходных данных.
- FFT Core: ядро БПФ. Содержит генератор поворачивающих коэффициентов, адресный генератор, блок управления и вычислительный блок.
- Memory Arbiter, Address Converter: выполняет арбитраж запросов к памяти со стороны интерфейса DMA_IF и ядра FFT. Также выполняет необходимые преобразования адресов.
- Memory Buffer 0/1: буферы памяти FFT. Размер каждого буфера – 4Kx64 бит.

22.6.3 Функционирование

22.6.3.1 Формат данных

В FFT используются следующие форматы данных:

- FP – 32-разрядная плавающая точка (стандарт IEEE-754);
- INT32 – 32-разрядное целое знаковое число (дополнительный код);
- INT16 – 16-разрядное целое знаковое число (дополнительный код).

Формат каждого 64-х разрядного слова входных данных в памяти FFT задается полем FORIN в регистре **FFT_CONF1**:

- 0 – комплексное число, FP
  {IMAG_FP$_0$, REAL_FP$_0$}
- 2 – комплексное число, INT16
  {IMAG_INT16$_1$, REAL_INT16$_1$, IMAG_INT16$_0$, REAL_INT16$_0$}
- 4 – вещественное число, FP
  {REAL_FP$_1$, REAL_FP$_0$}
- 6 – вещественное число, INT16
  {REAL_INT16$_3$, REAL_INT16$_2$, REAL_INT16$_1$, REAL_INT16$_0$}

Формат выходных данных задается полем FOROUT в регистре **FFT_CONF1**:

- 0 – FP
  {Y_FP$_1$, Y_FP$_0$}
- 1 – INT32
  {Y_INT32$_1$, Y_INT32$_0$}
- 2 – INT16
  {Y_INT16$_3$, Y_INT16$_2$, Y_INT16$_1$, Y_INT16$_0$}

Преобразование формата производится при загрузке данных во внутреннюю память FFT и при выгрузке из нее.

Формат внутренних вычислений: 32-разрядная плавающая точка.

Способ округления при вычислениях и при преобразовании форматов – к ближайшему числу, а при равноудаленности – к четному. При преобразовании FP→INT16 выполняется сатурация результата в диапазоне [-32768, 32767], а при преобразовании FP→INT32 – в диапазоне [-2^{32}, 2^{32} - 1].

22.6.3.2 Память БПФ-сопроцессора

Размер памяти БПФ-сопроцессора – 8K x 64 разрядных слов. Память расположена в общем пространстве памяти DSP – процессора.

Память может использоваться в 2-х буферном режиме, граница буферов соответствует середине памяти. Двухбуферный режим предназначен для непрерывного выполнения

преобразований в реальном времени. Он позволяет внешним устройствам (например, DSP процессору, DMA каналу, RISC ядру) параллельно с текущим преобразованием данных одного буфера работать с другим буфером памяти, например, выгрузить результаты предыдущего преобразования, загрузить данные для последующего преобразования. Максимальный размер преобразования в 2-х буферном режиме равен 4096. Номер буфера с преобразуемым массивом задается старшим битом начального адреса массива.

Если 2-х буферный режим не используется, то максимальный размер БПФ – 8192.

При выключенном БПФ-сопроцессоре (бит EN = 0 в регистре **FFT_CR**) вся его память доступна внешним устройствам.

Данные, подлежащие обработке, загружаются в БПФ-сопроцессор в прямом порядке. Результаты обработки выгружаются также в прямом порядке.

Действительным компонентам соответствуют нечетные адреса памяти, квадратурным (мнимым) – четные адреса.

22.6.3.3 Тригонометрические коэффициенты

Коэффициенты формируются во встроенных в FFT блоках. Обеспечивают как фазовые повороты выполняемого Фурье-преобразования, так и матричные фазовые повороты для матричного наращивания размера преобразования.

Предельный размер наращиваемого преобразования – 256K.

22.6.3.4 Производительность

За один такт выполняются максимально 40 арифметических операций с плавающей точкой (24 сложения / вычитания и 16 умножений). Например, при тактовой частоте 160 МГц производительность БПФ-сопроцессора составит 6400 MFLOP/c.

Время выполнения преобразования размера $N = 2^{2n}$: (N / 4 + 16)*n тактов.

Время выполнения преобразования размера $N = 2^{2n+1}$: (N / 4 + 16)*(n+1) тактов.

Сопутствующие операции (обратное преобразование, нормировка, фазовые матричные повороты, расчет мощностей, умножение результатов преобразования на комплексный массив) выполняются одновременно с основным преобразованием, без дополнительных затрат времени.

22.6.3.5 Выполняемые функции

22.6.3.5.1 Прямое быстрое преобразование Фурье

Размер комплексного массива $N = 2^n$, n = 4:13.

Порядок следования отсчетов на входе и на выходе – прямой. Логарифмический размер преобразования задается в регистре **FFT_CONF0** поле FT_LN = n.

22.6.3.5.2 Обратное быстрое преобразование Фурье

Параметры и режимы преобразования аналогичны прямому преобразованию. Режим задается битом FT_INV в регистре **FFT_CONF0**.

22.6.3.5.3 Нормировка результатов преобразования.

Мультипликативный коэффициент нормировки – константа в формате плавающей точки задается в регистре **FFT_NORC**. Нормировка выполняется как при обратном, так и при прямом преобразовании.

Режим задается битом FT_NOR в регистре **FFT_CONF0**.

22.6.3.5.4 Фазовые матричные повороты результатов преобразования

Предназначены для матричного наращивания размера преобразования до величины, превышающей 4096.

Предельный размер наращиваемого преобразования – 256K.

Выполняется как при прямом, так и при обратном преобразовании. Преобразуемый массив может быть только один. Логарифмические размеры строки и столбца матрицы наращивания задаются в регистре **FFT_CONF0** полями FT_LM и FT_LN соответственно, текущий номер преобразуемого столбца – в регистре **FFT_CONF0** полем FT_NCOL. Значения параметров FT_LM и FT_LN должны удовлетворять условию $13 \le FT\_LM + FT\_LN \le 18$.

Режим матричных поворотов задается битом FT_MX в регистре **FFT_CONF0**.

Алгоритм выполнения наращиваемого БПФ-преобразования может быть рассмотрен на примере преобразования размером 16384. Для выполнения такого преобразования удобно представить входной массив X[k], k=0, ..., 16383 в виде матрицы из 16-ти столбцов по 1024 элемента. Последовательность выполнения наращиваемого БПФ-преобразования в этом случае состоит из следующих шагов.

1) Настройка на обработку матрицы из 16-ти столбцов по 1024 элемента.

```
FFT_CONF0.FT_LN=0xA;          // размер столбца матрицы
                              // наращивания = 1024
FFT_CONF0.FT_LM=4;            // размер строки матрицы наращивания = 16
FFT_CONF0.FT_MX=1;            // режим матричного наращивания
FFT_CONF1.FT_ARRAY_NUM=0;     // число массивов = 1
```

2) Поочередное БПФ-преобразование 16-ти столбцов по 1024 элемента с последующим поворотом результатов преобразования в зависимости от номера текущего столбца.

```
for(j=0;j<16;j++) {                          // цикл по 16 столбцам
    FFT_CONF0.FT_NCOL=j;                     // номер столбца
    for(i=0;i<1024; ++) {X[i*16+j]➔ FFT}     // загрузка столбца
    FFT_CR.START=1;                          // пуск FFT
    while(FFT_SR.READY==0) {};               // ожидание конца FFT
    for(i=0;i<1024;i++) {X[i*16+j]➔ FFT }    // выгрузка столбца
}
```

3) Обработка строк по 16 элементов. Исходя из размера буфера (4096 элементов), удобно объединить строки в блоки по 256 строк.

```
FFT_CONF1.FT_ARRAY_NUM=0xFF;                 // число обрабатываемых
                                             // массивов (строк) = 256

for(j=0;j<4;j++) {                           //цикл по 4 блокам по 256 строк
    for(i=0;i<16*256;i++) {X[j*16*256+i]➔ FFT}   // загрузка блока из 256 строк
    FFT_CR.START=1;                          // пуск FFT
    while(FFT_SR.READY==0) {};               // ожидание конца FFT
    for(i=0;i<16*256;i++) {X[j*16*256+i]➔ FFT }  // выгрузка строки
}
```

Приведенный алгоритм не является единственно возможным. Он может быть оптимизирован исходя из задач пользователя.

22.6.3.5.5 Расчет мощностей результатов преобразования

На месте действительных компонент формируется сумма квадратов обеих компонент комплексных выходов, на месте мнимых размещаются нули. Выполняется как при прямом, так и при обратном преобразовании.

Режим задается битом FT_POW в регистре **FFT_CONF0**.

22.6.3.5.6 Одновременное преобразование M = 2m комплексных массивов

Размер каждого из преобразуемых массивов равен $N = 2^n$, $N \cdot M \leq 4096$.

Логарифмический размер каждого из обрабатываемых массивов задается параметром FT_LN. Количество обрабатываемых массивов задается параметром FT_ARRAY_NUM. Массивы размещаются в памяти последовательно, без разрывов.

Выполняются все сопутствующие операции, кроме матричного наращивания размера преобразования.

22.6.3.5.7 Поэлементное перемножение (с сопряжением) двух комплексных массивов

Один из массивов, а также и результирующий массив, расположен на месте преобразуемого массива, другой (сопрягаемый) – по адресу внутренней памяти БПФ-сопроцессора. Его начало должно быть выровнено по 256-разрядным словам.

Перемножаемые массивы должны располагаться в разных буферах внутренней памяти.

Логарифмический размер массивов задается параметром FT_LN.

Режим задается битом FT_MUL в регистре **FFT_CONF0**.

22.6.3.5.8 Быстрая БПФ – свертка (ковариация)

Последовательно при одном запуске БПФ-сопроцессора выполняются три процедуры:

- прямое БПФ;
- перемножение результата преобразования на комплексный массив частотной характеристики (с сопряжением последнего);
- обратное БПФ.

Логарифмический размер массивов задается параметром FT_LN.

Результат перемножения и выходной массивы располагаются на месте входного (преобразуемого) массива. Массив частотной характеристики располагается по адресу внутренней памяти БПФ-сопроцессора. Входной преобразуемый массив и массив частотной характеристики фильтра должны располагаться в разных буферах внутренней памяти БПФ-сопроцессора. Начало частотной характеристики должно быть выровнено по 256-разрядным словам. Входной преобразуемый массив может быть только один (M=1). Оба входных массива загружаются в прямом порядке, выходной выгружается также в прямом порядке.

Режим задается битом FT_CONV в регистре **FFT_CONF0**.

22.6.3.5.9 Вычисление скалярного произведения комплексных массивов

Выполняется поэлементное перемножение (с сопряжением) двух комплексных массивов, частичные произведения накапливаются в комплексном аккумуляторе (**FFT_ACC0**, **FFT_ACC1**).

Один из массивов расположен на месте преобразуемого массива, другой (сопрягаемый) – по адресу внутренней памяти БПФ-сопроцессора. Его начало должно быть выровнено по 256-разрядным словам. Перемножаемые массивы должны располагаться в разных буферах внутренней памяти.

Логарифмический размер массивов задается параметром FT_LN.

Режим задается битом FT_CMAC в регистре **FFT_CONF0**.

Выходные данные выдаются только в плавающей точке одинарной точности.

22.6.3.6 Особенности интерфейса DMA_IF

Доступ к памяти производится по 64-х разрядному интерфейсу DMA_IF, соответствующему спецификации AMBA AXI3. При этом допустимы обращения с размером 32 или 64 бита. Все остальные обращения приводят к ошибке на интерфейсе.

В интерфейсе не используется сигнал WSTRB, поэтому запись 32-х разрядного слова возможна только при соответствующем размере транзакции AWSIZE = 0x2.

22.6.4 Регистры

22.6.4.1 Перечень регистров

Ниже приведена сводная таблица доступных регистров блока FFT.

Поддерживаются только 32-х разрядные обращения к регистрам.

Таблица 22.89. Перечень регистров блока FFT (интерфейс CFG_IF)

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
FFT_CR	Регистр управления	0x0	WO	0x0000
FFT_SR	Регистр статуса	0x1	RO	0x0004
FFT_CONF0	Регистр конфигурации 0	0x0	RW	0x0008
FFT_CONF1	Регистр конфигурации 1	0x0	RW	0x000C
FFT_ADDRB	Регистр адреса входных данных массива 0	0x0	RW	0x0010
FFT_ADDRH	Регистр адреса входных данных массива 1	0x0	RW	0x0014
FFT_NORC	Регистр значения нормализации результата	0x0	RW	0x0018
FFT_IRQM	Регистр маски прерываний	0x0	RW	0x001C
FFT_IRQ	Регистр прерываний	0x0	RW	0x0020
FFT_ACC0	Регистр значения действительной части аккумулятора	0x0	RO	0x0024
FFT_ACC1	Регистр значения мнимой части аккумулятора	0x0	RO	0x0028

22.6.4.1.1 Регистр FFT_CR

Таблица 22.90. Формат регистра FFT_CR

Номер бита	Условное обозначение	Назначение
31:5	-	Резерв
4	ACC_CLR	Очистка значения аккумулятора
3	ACC_SET	Захват значения аккумулятора
2	START	Запуск обработки
1	CLR	Программный сброс
0	EN	Включение блока (влияет на работу конверторов адреса в память)

22.6.4.1.2 Регистр FFT_SR

Таблица 22.91. Формат регистра FFT_SR

Номер бита	Условное обозначение	Назначение
31:1	-	Резерв
0	READY	Готовность блока к работе

22.6.4.1.3 Регистр FFT_CONF0

Таблица 22.92. Формат регистра FFT_CONF0

Номер бита	Условное обозначение	Назначение
31:29	-	Резерв
28	FT_CMAC	Включение режима накопления произведения двух массивов

Номер бита	Условное обозначение	Назначение
27	FT_INH	Дополнительный массив с инверсной адресацией
26	FT_CONV	Включение режима свертки
25	FT_MUL	Включение режима умножения двух комплексных массивов
24:17	FT_NCOL	Номер столбца матричного преобразования минус один
16	FT_MX	Включение режима матричного наращивания
15	FT_NOR	Включение нормализации результата
14	FT_POW	Включение режима подсчета мощности
13	FT_INV	Включение режима инверсного преобразования
12:8	-	Резерв
7:4	FT_LM	Размер строки матричного преобразования (2^{FT_LM}). Выбирается так, чтобы выполнялось условие: $13 \leq FT\_LN + FT\_LM \leq 18$
3:0	FT_LN	Размер массива $N = 2^{FT\_LN}$ 0x0 – 0x3: резерв; 0x4: 16; 0x5: 32; … 0xD: 8192; 0xE, 0xF: резерв

22.6.4.1.4 Регистр FFT_CONF1

Таблица 22.93. Формат регистра FFT_CONF1

Номер бита	Условное обозначение	Назначение
31:14	-	Резерв
13:12	FOROUT	Формат выходных данных: 0x0:плавающая точка; 0x1: INT32; 0x2: INT16; 0x3: резерв
11:9	FORIN	Формат входных данных: 0x0: комплексное число, плавающая точка; 0x2: комплексное число, INT16; 0x4: вещественное число, плавающая точка; 0x6: вещественное число, INT16; 0x1, 0x3, 0x5, 0x7: резерв
8:0	FT_ARRAY_NUM	Количество обрабатываемых массивов минус один

22.6.4.1.5 Регистр FFT_ADDRB

Таблица 22.94. Формат регистра FFT_ADDRB

Номер бита	Условное обозначение	Назначение
31:12	-	Резерв
11:0	FT_ADDR_B	Начальный адрес основных входных данных

22.6.4.1.6 Регистр FFT_ADDRH

Таблица 22.95. Формат регистра FFT_ADDRH

Номер бита	Условное обозначение	Назначение
31:12	-	Резерв
11:0	FT_ADDR_H	Начальный адрес дополнительных входных данных

22.6.4.1.7 Регистр FFT_NORC

Таблица 22.96. Формат регистра FFT_NORC

Номер бита	Условное обозначение	Назначение
31:0	FT_NORC	Значение нормализации результата в формате плавающей точки

22.6.4.1.8 Регистр FFT_IRQM

Таблица 22.97. Формат регистра FFT_IRQM

Номер бита	Условное обозначение	Назначение
31:1	-	Резерв
0	RUN_DONE	Разрешение прерывания

22.6.4.1.9 Регистр FFT_IRQ

Регистр содержит статус прерывания. Для сброса прерывания необходимо записать '1' в соответствующий бит. Запись '0' игнорируется.

Таблица 22.98. Формат регистра FFT_IRQ

Номер бита	Условное обозначение	Назначение
31:1	-	Резерв
0	RUN_DONE	Статус прерывания

22.6.4.1.10 Регистр FFT_ACC0

Таблица 22.99. Формат регистра FFT_ACC0

Номер бита	Условное обозначение	Назначение
31:0	REAL	Значение действительной части аккумулятора

22.6.4.1.11 Регистр FFT_ACC1

Таблица 22.100. Формат регистра FFT_ACC1

Номер бита	Условное обозначение	Назначение
31:0	IMAG	Значение мнимой части аккумулятора

23. СОПРОЦЕССОР-УСКОРИТЕЛЬ GNSS

Четырех стандартное ГЛОНАСС/GPS/Galileo/BeiDou IP-ядро аппаратного ускорителя «Navicore-5L», разработки АО «НПЦ ЭЛВИС», встроенного в микросхему К1892ВМ21Я («Гиперком У»), обеспечивает высокоточный импульс в секунду точности «секундная метка»).

Сопроцессор-ускоритель позволит получать навигационное решение вместе с прошивкой навигационного ПО, поставляемого компанией «ЭЛВИС» для использования в любых связных системах и ТКО, а также, что особенно важно для реализации связных базовых станций на базе линейки «МАРКО-240».

Слежение выполняется программным обеспечением, выполняемым на базе 4-ядерного кластера центрального процессора с Arm v8 архитектурой (см. Главу 4), в соответствии с фактическим координатно-временным решением с целью достигнуть глобальной синхронизации PPS.

Приемник GNSS, построенный с использованием описываемого в настоящей главе сопроцессора-ускорителя GNSS предназначен для приема существующих и разрабатываемых перспективных систем спутниковой навигации.

Настоящая реализация приемника в чипе «Гиперком У» содержит только цифровые блоки аппаратной обработки сигнала и требует применения внешней аналоговой радио-части RFFE (Radio Frequency Front End), осуществляющей усиление, предварительную фильтрацию и оцифровку сигналов, принимаемых спутниковой антенной.

ПО приемника запускается на одном из процессорных ядер, входящих в состав микросхемы К1892ВМ21Я.

Концепция построения приемника GNSS показана на Рисунок 23.1.

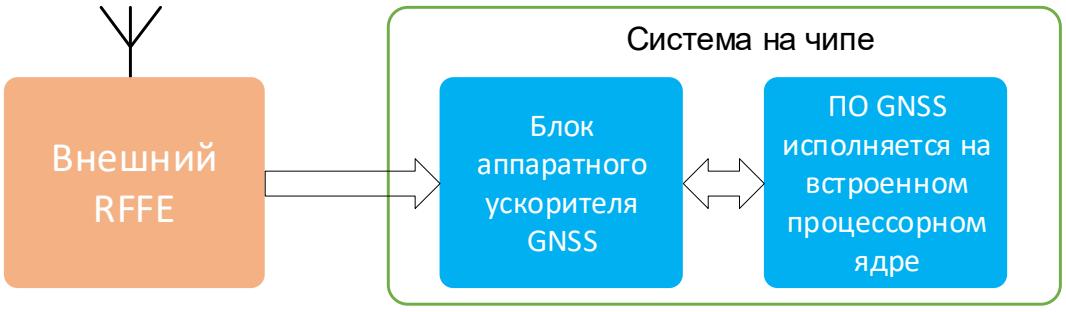


Рисунок 23.1. Общая схема построения приемника GNSS

Основной функциональной частью, обеспечивающей решение навигационных задач, является IP-ядро Navicore5L, являющееся аппаратным сопроцессором, осуществляющим многократные вычисления отсчетов ВКФ принимаемого и опорного сигналов, используемых в дальнейшем в программной реализации функций поиска и слежения. Блок-схема ускорителя представлена на следующем рисунке.

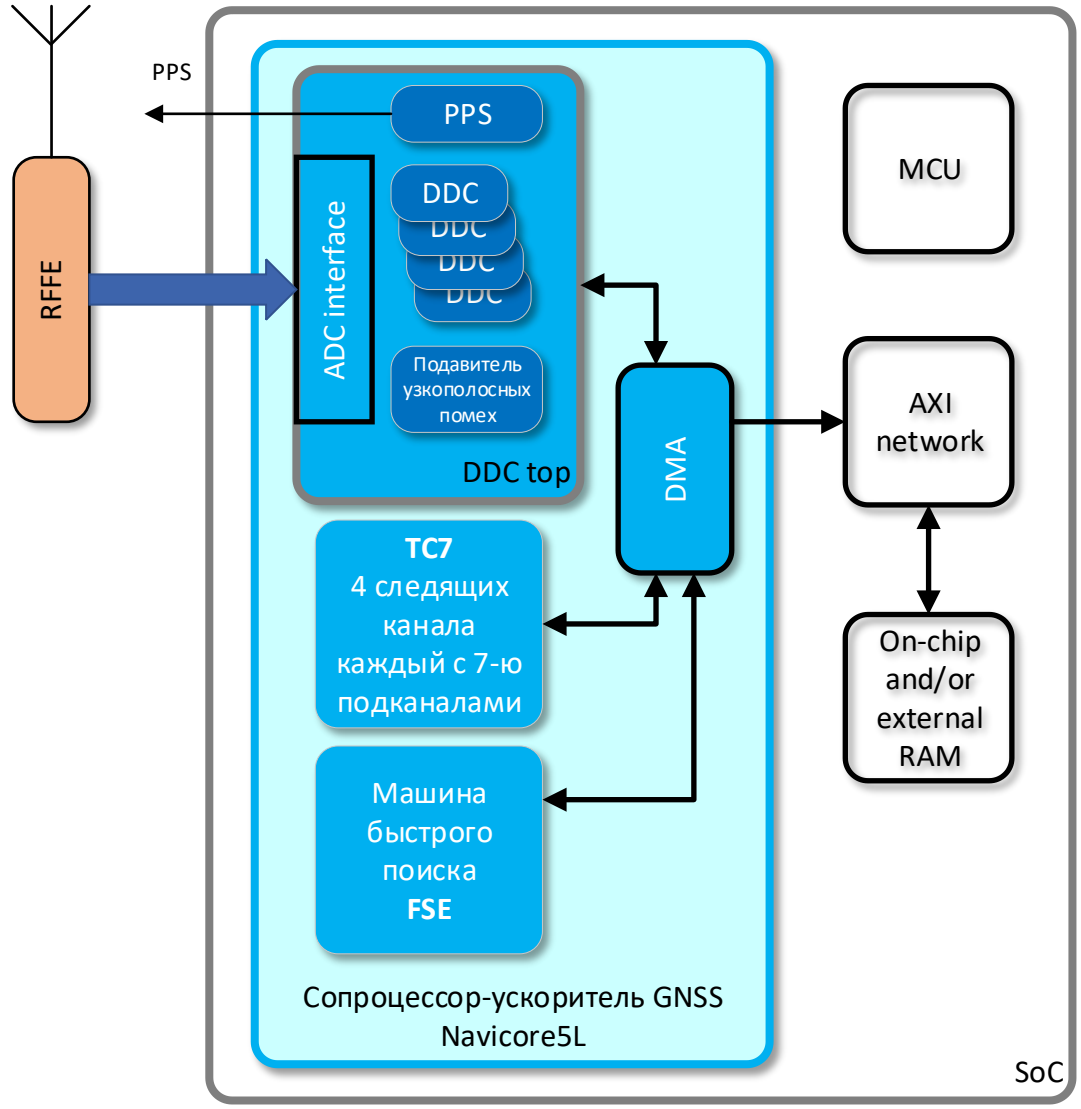


Рисунок 23.2. Блок-схема сопроцессора-ускорителя GNSS Navicore5L

Сопроцессор-ускоритель Navicore5L содержит 4 цифровых блока переноса частоты, фильтрации и децимации сигнала DDC (Digital Down Converter), блок прямого доступа к памяти DMA (Direct Memory Access), блок быстрого поиска сигнала FSE (Fast Search Engine), 4-канальный блок слежения с одновременным вычислением для 7 гипотез задержки TC7 (Tracking Channel), блок формирования секундной метки PPS (Pulse Per Second).

Приемник поддерживает следующие существующие системы кодов и диапазоны (частота в МГц) в режиме слежения:

- GPS открытые C/A, L1C и L2C коды диапазонов частот L1 (1575.42) и L2 (1227.6)
- GLONASS открытые CT коды диапазонов частот L1(1602+0.5625*k) и L2 (1246+0.4375*k)
- GALILEO открытый доступ в диапазоне E1 (1575.42)
- Beidou Phase II открытый доступ в диапазоне L1 (1561.098)

- GLONASS-K CDMA L1oc, L2oc и L3oc открытый доступ в диапазонах L1 (1600.995), L2 (1248.06) и L3 (1202.025)
- Beidou Phase III открытые коды L1C (1575.42) (в режиме загрузки в память, не тестировалось)
- Другие системы с произвольными кодами на базе комбинаций м-последовательностей длиной до 16383 бит, кодов Касами, последовательностей Лежандра.

Частоты диапазонов приведены для справки и задаются в RFFE при переносе на промежуточную частоту.

В режиме поиска приемник поддерживает следующие существующие системы кодов и диапазоны:

- GPS открытый код C/A, диапазонов частот L1 и L2.
- GLONASS открытый CT код диапазонов частот L1 и L2.
- GALILEO открытый доступ в диапазоне E1.
- Beidou Phase II открытый доступ в диапазоне L1.

Поддерживаются следующие виды модуляции:

- BPSK
- BOC (до 2-х поднесущих), в том числе:
- BOC1
- MBOC(1,6,1/11) (Galileo E1 OS)
- TMBOC(1,6,4/33) (GPS L1c, Beidou)
- Кодирование с почиповым мультиплексированием (GPS L2c, GLONASS-K CDMA L2oc)

Управление ускорителем осуществляется со стороны ПО, исполняющегося на одном из процессорных ядер, встроенных в систему на чипе, через регистры ICTL и специализированные порты для машины прямого доступа к памяти DMA.

23.1 Перечень регистров

Перечень программно-доступных регистров ICTL сопроцессора GNSS приведен в таблице 23.1.

Таблица 23.1. Регистры сопроцессора GNSS

Сме-щение	Условное обозначение	Дос-туп	Исходное состояние	Описание
0x0000-0x00FC	DDC_CH [0..3]			Регистры каналов DDC [0..3]
0x0000-0x003C	DDC_CH0			Регистры канала DDC0
0x0000	DDC_CH0_COMMON_CH_CTRL	RW	0x0285DF03	Режим работы канала
0x0004	DDC_CH0_DC_STATE	RW, RO	0x7F7F0000	Смещение нуля

Сме-щение	Условное обозначение	Дос-туп	Исходное состояние	Описание
0x0008	DDC_CH0_SCALE_SETUP	RW	0x00000000	Установка режима масштабирования в блоках
0x000C	DDC_CH0_BWG_ESTM	RW	0x00000000	Оценка уровня сигнала
0x0010	DDC_CH0_HTRD_FRQ	RW	0x00000000	Частота гетеродина
0x0014-0x001C	Резерв			
0x0020	DDC_CH0_DFIR32_COFF0	RW	0x00000000	Коэффициенты фильтра
0x0024	DDC_CH0_DFIR32_COFF1	RW	0x00000000	Коэффициенты фильтра
0x0028	DDC_CH0_DFIR32_COFF2	RW	0x00000000	Коэффициенты фильтра
0x002C	DDC_CH0_DFIR32_COFF3	RW	0x00000000	Коэффициенты фильтра
0x0030	DDC_CH0_DFIR32_COFF4	RW	0x00000000	Коэффициенты фильтра
0x0034	DDC_CH0_DFIR32_COFF5	RW	0x00000000	Коэффициенты фильтра
0x0038	DDC_CH0_DFIR32_COFF6	RW	0x00000000	Коэффициенты фильтра
0x003C	DDC_CH0_DFIR32_COFF7	RW	0x00000000	Коэффициенты фильтра
0x0040-0x007C	DDC_CH1			Регистры канала DDC1
0x0080-0x00BC	DDC_CH2			Регистры канала DDC2
0x00C0-0x00FC	DDC_CH3			Регистры канала DDC3
0x0100-0x017C	Резерв			
0x0180-0x01FC	DDC_TOP			Общие регистры управления DDC
0x0180	DDC_TOP_MODE	RW	0x02AA0000	Основной регистр уравления блоком
0x0184	DDC_TOP_FIFO_TRS	RW	0x00000000	Пороги срабатывания флагов "почти полное fifo"
0x0188	DDC_TOP_FIFO_STATE	RO	0x00421084	Регистр состояний fifo
0x018C-0x01FC	Резерв			
0x0200-0x027C	DDC_PPS			Регистры блока PPS
0x0200	DDC_PPS_POS_MS	RW	0xFFFFFFFF	Позиция пульса PPS относительно локальной миллисекунды, выраженная в тактах
0x0204	DDC_PPS_POS0	RO	0xFFFFFFFF	Позиция пульса PPS относительно начала блока данных соответствующего канала, выраженная в тактах ADC_CLK 0
0x0208	DDC_PPS_POS1	RO	0xFFFFFFFF	Позиция пульса PPS относительно начала блока данных соответствующего канала, выраженная в тактах ADC_CLK 1
0x020C	DDC_PPS_POS2	RO	0xFFFFFFFF	Позиция пульса PPS относительно начала блока данных соответствующего канала, выраженная в тактах ADC_CLK 2
0x0210	DDC_PPS_POS3	RO	0xFFFFFFFF	Позиция пульса PPS относительно начала блока данных соответствующего канала, выраженная в тактах ADC_CLK 3

Сме-щение	Условное обозначение	Дос-туп	Исходное состояние	Описание
0x0214	DDC_PPS_BLOCK_CNT0	RO	0xFFFFFFFF	Количество сформированных блоков данных с момента старта 0
0x0218	DDC_PPS_BLOCK_CNT1	RO	0xFFFFFFFF	Количество сформированных блоков данных с момента старта 1
0x021C	DDC_PPS_BLOCK_CNT2	RO	0xFFFFFFFF	Количество сформированных блоков данных с момента старта 2
0x0220	DDC_PPS_BLOCK_CNT3	RO	0xFFFFFFFF	Количество сформированных блоков данных с момента старта 3
0x0224	DDC_PPS_MS_CNT	RO	0x00000000	Счётчик миллисекунд
0x0228	DDC_PPS_BLOCK_SIZE0	RW	0x00F42400	Размер блока данных соответствующего канала в отсчётах 0
0x022C	DDC_PPS_BLOCK_SIZE1	RW	0x00F42400	Размер блока данных соответствующего канала в отсчётах 1
0x0230	DDC_PPS_BLOCK_SIZE2	RW	0x00F42400	Размер блока данных соответствующего канала в отсчётах 2
0x0234	DDC_PPS_BLOCK_SIZE3	RW	0x00F42400	Размер блока данных соответствующего канала в отсчётах 3
0x0238	DDC_PPS_CLK_DIV	RW	0x00F42400	Основной делитель тактовой частоты для PPS
0x023C	DDC_PPS_MS_DIV	RW	0x00003E80	Основной делитель тактовой частоты для миллисекунды
0x0240	DDC_PPS_FINE_TUNE	RW	0x00000000	Маска для точной подстройки делителя
0x0244	DDC_PPS_CLK_PH_ADJ	RW	0x00000000	Подстройка фазы PPS выражаемая в поправке к CLK_DIV для подстройки фазы PPS в процессе работы
0x0248	DDC_PPS_DELAY	RW	0x00000000	Задержка выходного сигнала PPS_OUT
0x024C	DDC_PPS_WIDTH	RW	0x80003E80	Фаза и длительность PPS_OUT
0x0250-0x027C	Резерв			
0x0280-0x02FC	DDC_ADC			Регистры управления интерфейсом с АЦП
0x0280	DDC_ADC_LUT_DATA0	RW	0x00000000	Таблица преобразования входных данных 0
0x0284	DDC_ADC_LUT_DATA1	RW	0x00000000	Таблица преобразования входных данных 1
0x0288	DDC_ADC_LUT_DATA2	RW	0x00000000	Таблица преобразования входных данных 2
0x028C	DDC_ADC_LUT_DATA3	RW	0x00000000	Таблица преобразования входных данных 3
0x0290	DDC_ADC_CTRL	RW	0x00000000	Режим работы каналов данных АЦП
0x0294-0x02FC	Резерв			

Сме-щение	Условное обозначение	Дос-туп	Исходное состояние	Описание
0x0300-0x037C	DDC_JR			Регистры подавителя помех
0x0300	DDC_JR_COMMON_CTRL	RW	0x00040030	Регистр управления\состояния подавителя
0x0304	DDC_JR_OP_CTRL	RW	0x00000080	Регистр оперативного управления
0x0308	DDC_JR_CIC_CTRL	RW	0x000000FF	Регистр управления CIC каскадами децимации\интерполяции
0x030C	DDC_JR_FIR_CTRL	RW	0x0000071F	Регистр управления FIR каскадом
0x0310	DDC_JR_DLY_CTRL	RW,	0x00000000	Регистр управления линией задержки
0x0314	DDC_JR_FGAIN	RW	0x00000798	Регистр управления усилением в каналах подавителя (старшая часть)
0x0318-0x031C	Резерв			
0x0320	DDC_JR_FRQ0	RW	0x00000000	Регистр частоты канала 0
0x0324	DDC_JR_FRQ1	RW	0x00000000	Регистр частоты канала 1
0x0328	DDC_JR_FRQ2	RW	0x00000000	Регистр частоты канала 2
0x032C	DDC_JR_FRQ3	RW	0x00000000	Регистр частоты канала 3
0x0330	DDC_JR_FRQ4	RW	0x00000000	Регистр частоты канала 4
0x0334	DDC_JR_FRQ5	RW	0x00000000	Регистр частоты канала 5
0x0338	DDC_JR_FRQ6	RW	0x00000000	Регистр частоты канала 6
0x033C	DDC_JR_FRQ7	RW	0x00000000	Регистр частоты канала 7
0x0340	DDC_JR_FIR_COFF0	RW	0x00000000	Регистр коэффициентов FIR фильтра 0
0x0344	DDC_JR_FIR_COFF1	RW	0x00000000	Регистр коэффициентов FIR фильтра 1
0x0348	DDC_JR_FIR_COFF2	RW	0x00000000	Регистр коэффициентов FIR фильтра 2
0x034C	DDC_JR_FIR_COFF3	RW	0x00000000	Регистр коэффициентов FIR фильтра 3
0x0350	DDC_JR_FIR_COFF4	RW	0x00000000	Регистр коэффициентов FIR фильтра 4
0x0354	DDC_JR_FIR_COFF5	RW	0x00000000	Регистр коэффициентов FIR фильтра 5
0x0358	DDC_JR_FIR_COFF6	RW	0x00000000	Регистр коэффициентов FIR фильтра 6
0x035C	DDC_JR_FIR_COFF7	RW	0x00000000	Регистр коэффициентов FIR фильтра 7
0x0360-0x07FC	Резерв			
0x0800-0x083C	TC7			Регистры TC7
0x0800	TC7_CTRL	WC, RW	0x00000000	Регистр управления
0x0804	TC7_STATUS	RO	0x00000300	Регистр статуса
0x0808	TC7_READ_GATE	RW	0x00000000	Запись адреса встроенной памяти TC7, чтение значения по записанному адресу
0x080C	TC7_WRITE_GATE	RW	0x00000000	Последовательная запись встроенной памяти
0x0810-0x0FFC	Резерв			

Сме-щение	Условное обозначение	Дос-туп	Исходное состояние	Описание
0x1000-0x1004	FSE			Регистры FSE
0x1000	FSE_CTRL	WC, RW	0x00000000	Регистр управления
0x1004	FSE_STATUS	RO	0x00000000	Регистр статуса
0x1008-0x27FC	Резерв			
0x2800-0x28FC	DMA			Регистры DMA
0x2800	DMA_CH_0_CSR_0	RW, RO	0x00000000	Регистр управления/статуса канала
0x2804	DMA_CH_1_CSR_0	RW, RO	0x00000000	Регистр управления/статуса канала
0x2808	DMA_CH_2_CSR_0	RW, RO	0x00000000	Регистр управления/статуса канала
0x280C	DMA_CH_3_CSR_0	RW, RO	0x00000000	Регистр управления/статуса канала
0x2810	DMA_CH_4_CSR_0	RW, RO	0x00000000	Регистр управления/статуса канала
0x2814	DMA_CH_5_CSR_0	RW, RO	0x00000000	Регистр управления/статуса канала
0x2818	DMA_CH_6_CSR_0	RW, RO	0x00000000	Регистр управления/статуса канала
0x281C	DMA_CH_7_CSR_0	RW, RO	0x00000000	Регистр управления/статуса канала
0x2820	DMA_CH_0_CSR_1	RW	0x00000000	Регистр управления/статуса канала
0x2824	DMA_CH_1_CSR_1	RW	0x00000000	Регистр управления/статуса канала
0x2828	DMA_CH_2_CSR_1	RW	0x00000000	Регистр управления/статуса канала
0x282C	DMA_CH_3_CSR_1	RW	0x00000000	Регистр управления/статуса канала
0x2830	DMA_CH_4_CSR_1	RW	0x00000000	Регистр управления/статуса канала
0x2834	DMA_CH_5_CSR_1	RW	0x00000000	Регистр управления/статуса канала
0x2838	DMA_CH_6_CSR_1	RW	0x00000000	Регистр управления/статуса канала
0x283C	DMA_CH_7_CSR_1	RW	0x00000000	Регистр управления/статуса канала
0x2840	DMA_IRQ_CSR	RW, RO	0x00000000	Регистр управления/статуса прерываний
0x2844-0x285C	Резерв			
0x2860	DMA_CH_0_CSR_2	RW	0x0000001F	Регистр управления/статуса канала
0x2864	DMA_CH_1_CSR_2	RW	0x0000001E	Регистр управления/статуса канала
0x2868	DMA_CH_2_CSR_2	RW	0x0000001D	Регистр управления/статуса канала
0x286C	DMA_CH_3_CSR_2	RW	0x0000001C	Регистр управления/статуса канала
0x2870	DMA_CH_4_CSR_2	RW	0x0000001B	Регистр управления/статуса канала
0x2874	DMA_CH_5_CSR_2	RW	0x0000001A	Регистр управления/статуса канала

Сме-щение	Условное обозначение	Дос-туп	Исходное состояние	Описание
0x2878	DMA_CH_6_CSR_2	RW	0x00000028	Регистр управления/статуса канала
0x287C	DMA_CH_7_CSR_2	RW	0x00000080	Регистр управления/статуса канала
0x2880-0x2FFC	Резерв			
0x3000-0x300C	TST_DATA_UNIT			Регистры управления TST_DATA_UNIT
0x3000	TST_DATA_UNIT_CTRL	RW,	0x04000000	Основной регистр управления блоком
0x3004	TST_DATA_UNIT_AXI_MODE	RW	0x1F000000	Регистр управления для режима с подачей данных от DMA
0x3008	TST_DATA_UNIT_SRC_HI	RW	0x80808080	Управление источником данных
0x300C	TST_DATA_UNIT_SRC_LO	RW	0x80808080	Управление источником данных
0x3010-0x37FC	Резерв			
0x3800-0x385C	CTRL			Общие регистры управления
0x3800	CTRL_VERSION		0x00000500	Регистр версии
0x3804-0x3808	Резерв			
0x380C	CTRL_TEST_RW	RW	0x00001234	Регистр для проверки чтения/записи
0x3810-0x383C	Резерв			
0x3840	CTRL_CMP_PATTERN_L	RW	0x00000000	Младшая часть начального состояния для проверки тестовой последовательности
0x3844	CTRL_CMP_PATTERN_H	RW	0x00000001	Старшая часть начального состояния для проверки тестовой последовательности
0x3848	CTRL_CMP_ERROR_CNT	RO	0x00000000	Счетчик ошибок
0x384C	CTRL_GEN_PATTERN_L	RW	0x00000000	Младшая часть начального состояния генератора тестовой последовательности
0x3850	CTRL_GEN_PATTERN_H	RW	0x00000001	Старшая часть начального состояния генератора тестовой последовательности
0x3854-0x3FFC	Резерв			

23.2 Формат упаковки данных

GNSS приемник использует общую память системы на чипе для хранения цифрового представления сигналов, используемых для решения навигационной задачи. В штатном режиме блоки данных формируются с помощью устройств, входящих в блок DDC_TOP, как результат обработки сигналов, принимаемых с аналоговой части приемника и

оцифрованных с помощью АЦП (не входят в состав СнЧ). Машина DMA записывает блоки данных в память для временного хранения. Затем та же машина DMA осуществляет многократную загрузку блоков данных в модули слежения ТС7 и поиска FSE. В отладочном режиме допустимо формирование данных в памяти искусственным способом (с помощью ПО) или загрузка готовых данных.

Данные, подлежащие обработке блоками FSE и ТС7, организованы в виде фреймов из 64-битных слов. Первое слово фрейма содержит описание данных, содержащихся во фрейме и служебную информацию. В частности, в заголовке содержится информация о количестве отсчетов и их разрядности, что позволяет определить размер фрейма. Структура заголовка фрейма приведена в следующей таблице.

Таблица 23.2. Структура заголовка фрейма сигнала

Биты	Название	Пояснение
2..0	FRM_STREAM	Номер потока обработки. Не оказывает влияния на обработку. Предназначен для пометки выходных данных – результатов поиска. Значения 0-3 соответствуют каналам модуля DDC. Значения 4-7 применяются при формировании блока средствами, отличными от DDC. Применяются, например, для отладки и тестирования.
3	–	Резерв
6..4	FRM_IN_FMT	Формат отсчетов данных: 0: 2x1 IQ – знак (mid-rizer); 1: 2x2 IQ – знак+амплитуда (mid-rizer); 2: 2x4 IQ – двоично-дополнительный симметрично ограниченный -7<x<7; 3: 2x8 IQ – двоично-дополнительный симметрично ограниченный -127<x<127; 4 - 6: запрещенные значения; 7: не используется, зарезервировано для дальномерного кода
7	CHIP_SYN	Бит индицирует чип-синхронную дискретизацию. 0: дискретизация на «высокой» частоте, несинхронной с частотой чипов. Фрейм может использоваться как для слежения в ТС7, так и для поиска в FSE. 1: дискретизация на «низкой» частоте, кратной чиповой. Фрейм может использоваться только для поиска в FSE.
23..8	FRM_COUNT	Номер фрейма. Значение счетчика фреймов в момент формирования данного фрейма.
39..24	FRM_NSAMP	Количество отсчетов во фрейме, уменьшенное на 1
63..40	–	Резерв

Сразу после заголовка следуют слова, содержащие отсчеты сигнала. Их количество во фрейме определяется как произведение (FRM_NSAMP-1) на полную разрядность отсчета, деленное на 64 и округленное в большую сторону. Последнее во фрейме слово данных может содержать неполное количество отсчетов. В этом случае, старшие биты слова игнорируются блоками ТС7 и FSE.

Отсчеты кодовой последовательности (при FRM_IN_FMT==7) упакованы в слова таким образом, что в младших битах слова содержатся ранние по времени отсчеты.

Способы упаковки комплексных отсчетов сигнала (при FRM_IN_FMT от 0 до 3) представлены ниже.

Таблица 23.3. Упаковка комплексных отсчетов сигнала

Компонента	Расположение в разрядах слова в зависимости от формата			
	0: 2x1 mid-rizer	1: 2x2 mid-rizer	2: 2x4 two's compl.	3: 2x8 two's compl.
Отсчет 0, Q	0	1..0	3..0	7..0
Отсчет 0, I	1	3..2	7..4	15..8
Отсчет 1, Q	2	5..4	11..8	23..16
Отсчет 1, I	3	7..6	15..12	31..24
Отсчет 2, Q	4	9..8	19..16	39..32
Отсчет 2, I	5	11..10	23..10	47..40
…	…	…	…	…

При обработке в модулях TC7 и FSE все форматы данных предварительно приводятся к внутреннему двоично-дополнительному симметрично ограниченному формату. Преобразование осуществляется по таблице.

Таблица 23.4. Преобразование комплексных отсчетов входных сигналов в TC7 и FSE

0: 2x1 знак (mid-rizer)		1: 2x2 знак + амплитуда (mid-rizer)		2: 2x4 двоично-дополнительный	
поле	значение	поле	значение	поле	значение
0	4	01	6	0111	7
1	-4	00	2	0110	6
		10	-2	0101	5
		11	-6	0100	4
				0011	3
				0010	2
				0001	1
				0000	0
				1111	-1
				1110	-2
				1101	-3
				1100	-4
				1011	-5
				1010	-6
				1001	-7
				1000	-7

23.3 Блок управления интерфейсами и DDC – DDC_TOP

Модуль DDC_TOP объединяет в себе несколько блоков, относящихся к входным интерфейсам подсистемы GNSS. В нем осуществляется:

- общее управление приемом потоков данных с внешних АЦП,
- предварительная обработка данных (гетеродинирование, децимация и фильтрация) с помощью DDC,

- упаковка выходных данных DDC в 64-битные слова для передачи по 64 битному DMA интерфейсу,
- формирование сигнала PPS.

Блок-схема приведена на изображении ниже.

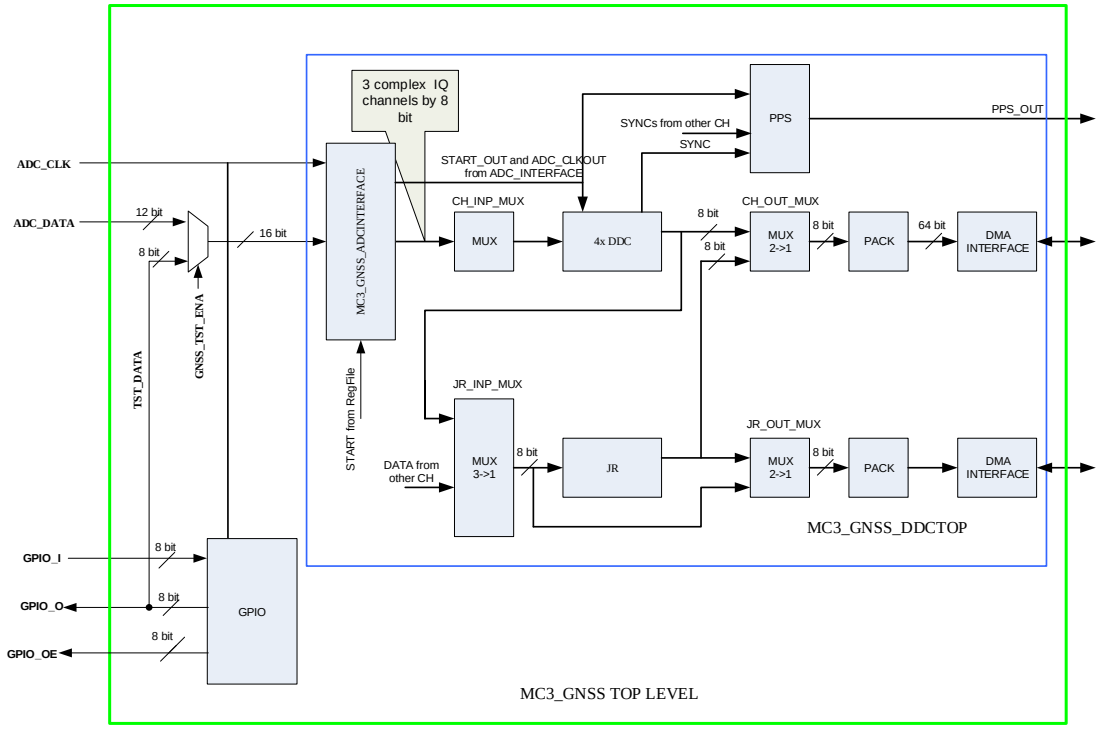


Рисунок 23.3. Блок-схема модуля DDC_TOP

Сигналы ADC_DATA[11.0] на входе в блок формируются преобразователями уровня из внешних сигналов, подаваемых на 2 группы контактов микросхемы. Одна из групп предназначена для сигналов низкого уровня (LVCMOS, 1.8v) и преобразовывается в сигнал ADC_DATA_LV[11..0], другая – для сигналов высокого уровня (HVCMOS, 3.3v), ADC_DATA_HV[11..0]. Выбор между ними осуществляется записью нуля или единицы в поле ADC_LVL_SEL регистра DDC_TOP_MODE (см. описание регистров).

Блок DDC_TOP объединяет несколько подблоков, являющихся самостоятельными устройствами и осуществляет общее управление ими и синхронизацию.

MC3_GNSS_ADCINTERFACE принимает с 12 разрядной шины данных ADC_DATA потоковые данные, сопровождаемые тактовым сигналом ADC_CLK. В своем составе ADC_INTERFACE имеет на входе инвертер и регулируемую линию задержки для тактового сигнала с шагом настройки ~100pS, предназначенную для компенсации задержки формирования данных на выходе АЦП. С выхода этой линии задержки тактовый сигнал поступает в глобальную тактовую сеть ADC_CLK_G, и используется всеми блоками, работающими с частотой ADC_CLK. Подробнее блок описан в Разделе 23.4.

DDC осуществляет гетеродинирование, децимацию и фильтрацию сигнала. В составе DDC_TOP имеется 4 канала DDC. На блок-схеме показано подключение одного канала.

На входе каждого канала DDC стоит мультиплексор, который позволяет подключить канал к любому из четырех выходных каналов MC3_GNSS_ADCINTERFACE. Конфигурация подключения настраивается однократно перед стартом и более не изменяется.

Синхронный старт всего блока производится записью «1» поле DDC_ENA регистра DDC_TOP_MODE. Это поле непосредственно управляет входным асинхронным сигналом старта START_IN блока MC3_GNSS_ADCINTERFACE и разрешает его работу. Подробнее блок описан в Разделе 23.5.

Выход каждого DDC подключен к своему упаковщику данных PACK и далее к индивидуальному DMA интерфейсу.

JR осуществляет подавление узкополосных помех на выходе каналов DDC_CHN и может подключаться к выходу любого канала в режиме подмены выходного потока канала или в режиме формирования отдельного потока данных. Подробнее блок описан в Разделе 23.6. Модуль имеет на выходе свой собственный упаковщик и DMA интерфейс.

PACK осуществляет упаковку выходных данных каналов в 64 битные слова, для передачи по 64-разрядной шине DMA.

DMA INTERFACE буферизирует потоковые данные в FIFO и поставляет упакованные данные для записи через AXI во внешнюю память. Размер буфера (глубина FIFO) 32 слова.

DMA и упаковщик сбрасываются в исходное состояние с очисткой флагов ошибок в регистровом файле сигналом JR_ENA или CH_ENA в зависимости от того к какому поставщику подключена эта связка. То есть при выключении источника данных связка приводится в исходное состояние.

PPS формирует сигнал Pulse Per Second.

23.4 Интерфейс с АЦП – MCC3_DDC_ADCINTERFACE

Блок предназначен для приема потока данных от АЦП и формирования потоков данных для блоков приемной части Navicore5L. Он имеет следующие основные характеристики:

- Кол-во каналов АЦП – до 6
- Кол-во Front-End – до 4
- Разрядность данных 2/3/4/6/8 bit
- Стандарты интерфейса
  - SDR parallel
  - DDR parallel
  - JESD207 Dual Port Full Duplex 8-bit SDR/DDR.
- Производительность до 250 MSPS.

Схема обработки сигнала модулем представлена на рисунке

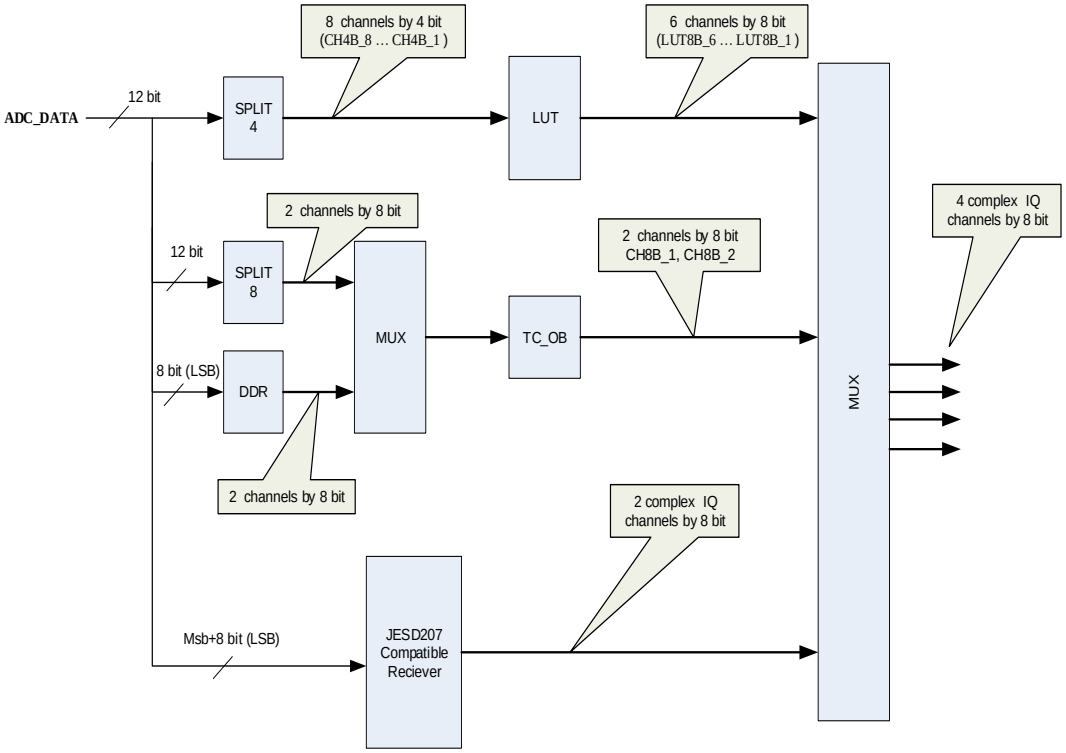


Рисунок 23.4. Преобразование сигнала в модуле интерфейса АЦП

Интерфейс с АЦП осуществляет преобразование (расщепление) шины данных АЦП или потока данных JESD207 в несколько внутренних потоков данных соответствующих разным каналам АЦП, преобразование отсчетов во внутренний формат 2x8 отсчетов на канал и общую синхронизацию системы.

Блок работает в режиме простой параллельной шины данных (BUS_MODE=0) или в режиме совместимости с одним из режимов работы JESD207 интерфейса (BUS_MODE=1).

Входной поток данных блока представляет собой 12-разрядную шину данных ADC_DATA с тактовым сигналом ADC_CLK, подключенную к внешним контактам СнЧ. В зависимости от настроек DDC_TOP, выбираются внешние контакты, которые могут принимать сигналы высокого (3.3 В) или низкого (1.8 В) логического уровня. Шина может работать в режиме SDR с частотой выборки до 250 MSpS, в режиме DDR с частотой выборки до 150 MSpS или в режиме JESD207.

Таким образом, перекрываются диапазоны частот дискретизации (тактовой частоты интерфейса) по выборке скоростных АЦП от Analog Devices:

	100MSPS	150MSPS	250MSPS
LVCMOS SDR			
LVCMOS DDR			

Выходной поток данных – 4 квадратурных канала данных CH1 – CH4 разрядностью 8 бит на квадратуру (всего 16 бит), синхронных с тактовым сигналом ADC.

23.5 Преобразователи частоты и дециматоры DDC0 – DDC3

Navicore5L включает в себя 4 идентичных модуля DDC: DDC0 – DDC3.

Блок предназначен для преобразования частоты сигнала (гетеродинирования), его фильтрации и децимации. Он обладает следующими основными характеристиками:

- Частота дискретизации входного сигнала до: 250MSPS;
- Входной сигнал: аналитический (квадратурный) или действительный цифровой сигнал с разрядностью до 8 бит на компоненту;
- Цифровой квадратурный гетеродин, реализованный по алгоритму CORDIC;
- Коэффициент децимации: регулируемый от 1 до 3*2*2=12

Блок работает в двух тактовых доменах – домене частот ADC_CLK (входные сигналы с АЦП) и системном домене (выходные FIFO).

Входной поток данных представляет собой две 8 разрядные шины данных, синхронные с тактовым сигналом ADC_CLK. Данные поступают каждый такт.

Выходной поток данных имеет такой же формат – две 8 разрядные шины данных, но синхронизирован с тактовым сигналом PCLK, имеющим более высокую частоту, чем ADC_CLK, и сопровождается сигналом валидности данных.

Сигнал RESET сбрасывает все регистры, конвейеры и регистровый файл в исходное состояние, запрещает работу блоков, запрещает запись в регистровый файл.

Сигнал START имеет активный уровень единица и устанавливается по возрастающему фронту ADC_CLK. При START=0 блокируется работа, очищаются конвейеры и линии задержки, счетчики децимации и разрешается запись в регистровый файл – управление DDC разблокируется. При START=1 разрешается обработка, но блокируется управление.

Регистры DC_STATE и SCALE_SETUP не блокируются никогда.

В следующих разделах описаны регистры управления каналами DDC0 — DDC3 на примере канала DDC0. Управление всеми каналами DDC осуществляется одинаково, за исключением смещения регистров.

23.6 Подавитель узкополосных помех (JR)

Блок JR является подавителем узкополосных помех компенсационного типа. Подавитель содержит 8 каналов, выделяющих 8 узкополосных помех для последующего их вычитания из полезного сигнала. Для компенсации задержки, возникающей в каналах выделения помехи, входной сигнал также задерживается с помощью линии задержки, входящей в состав модуля. Блок имеет вход и выход, представляющие собой шины данных одинаковой разрядности 2x8 бит каждая. На вход подается комплексный сигнал с выхода выбранного DDC. С выхода сигнал подается на упаковщик выбранного канала DDC или собственный упаковщик блока JR.

23.7 Контроллер DMA

Многоканальный контроллер прямого доступа к памяти осуществляет все основные операции, связанные с передачей потоков данных между устройствами корреляционного сопроцессора и памятью. Передача данных осуществляется с использованием шины AXI с помощью нескольких независимых каналов с разделением по времени, согласованием последовательности выполнения и приоритетами.

DMA контроллер имеет следующие возможности:

- Передача по 8 независимым каналам с разделением по времени;
- Программная конфигурация пропускной способности по шине AXI для каждого канала (установка доступной глубины очереди адресных запросов);
- Механизм межканального взаимодействия для согласования последовательности выполнения задач;
- Выполнение связанного списка задач;
- 2 уровня приоритета для передачи и 2 уровня приоритета для приема данных;
- Обработка заданий по флагам готовности каналов данных;
- Конвейеризация адресной очереди и раздельная обработка флагов завершения фаз данных для максимальной производительности;

Задачи, связанные с записью в память потоков данных от АЦП и формированием непрерывного потока вывода, выполняются с высоким уровнем приоритета, в то время как задачи по передаче данных на обработку – с низким. Это позволяет избежать пропусков приема или передачи непрерывных потоков данных.

Программирование контроллера осуществляется путем задания цепочек дескрипторов заданий для контроллера в памяти. Контроллер выполняет их последовательно, тем самым осуществляя реализацию предопределенного алгоритма обработки. При этом цепочки данных низкого приоритета могут прерываться на время выполнения высокоприоритетных задач.

23.8 Блок следящих каналов TC7

Блок корреляционной обработки слежения TC7 (Tracking channels with 7 subchannels) предназначен для ускорения вычислений, осуществляемых при реализации алгоритмов слежения за временем задержки и частотой Доплера широкополосных сигналов GNSS. Общая схема блока приведена на следующем рисунке.

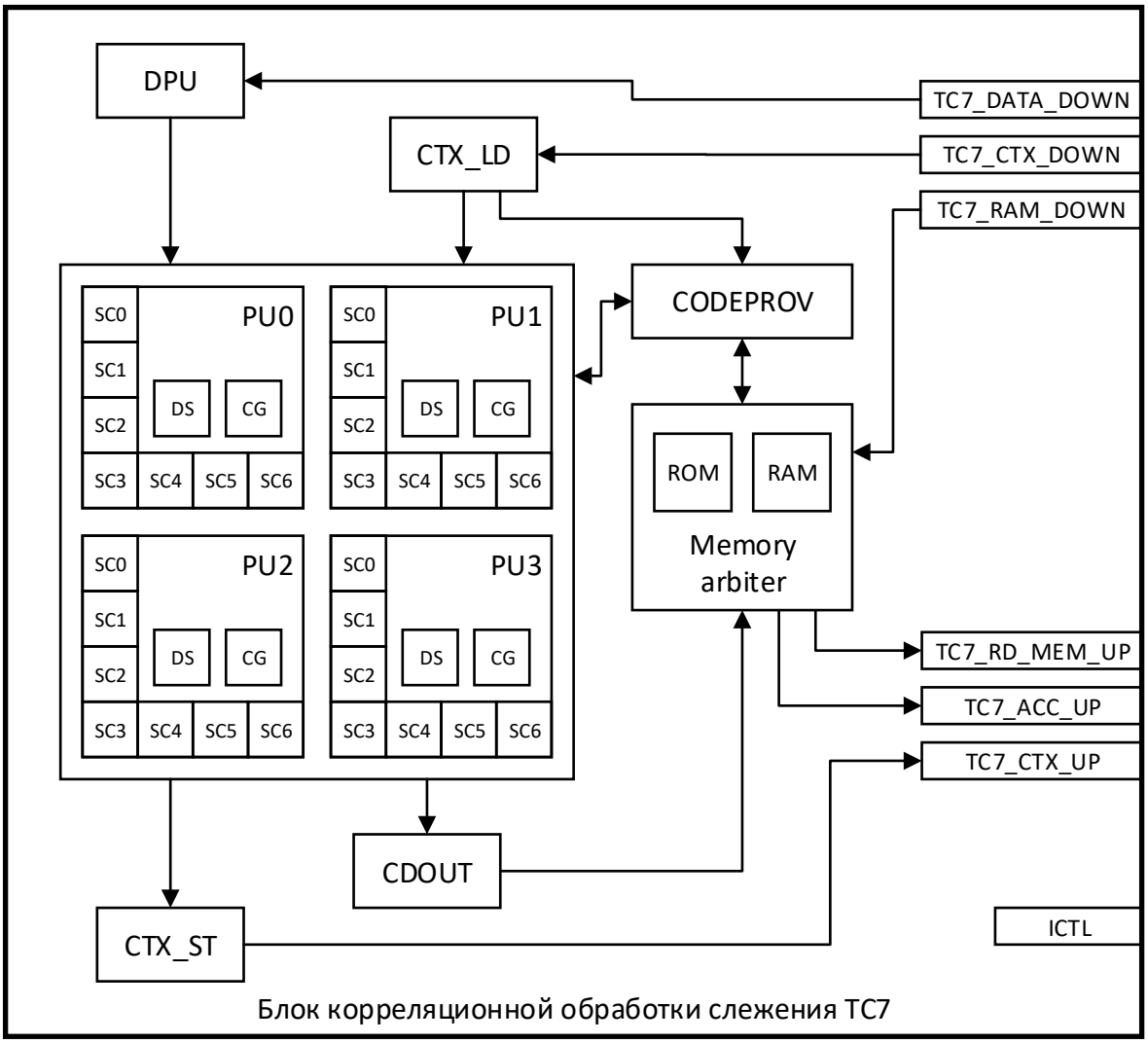


Рисунок 23.5. Схема блока следящих каналов TC7

Он включает в себя блоки интерфейсов данных и управления (порты DMA, регистры ICTL), блок подготовки данных (DPU – Data Prepare Unit), блок управления выводом (CDOUT – Correlation Data Output), блок формирования кодов (CODEPROV – Code Provider), блоки загрузки и сохранения контекста (CTX_LD и CTX_ST) и 4 блока обработки (PU – Processing Unit). Каждый PU имеет в своем составе 7 подканалов-корреляторов (SC – Sub-Channel), дециматор (DS – Down-Sampler) и генератор кодов (CG – Code Generator).

Также в модуль включены блоки памяти – ROM со стандартными кодами навигационных сигналов и RAM для результатов накопления и нестандартных кодов.

Основным блоком обработки данных является блок устройств обработки (Processing Unit – PU). В нем осуществляются основные физические преобразования сигнала:

- вычисление оценок взаимной корреляции между входным сигналом и одной или двумя опорными кодовыми последовательностями при разных (до 7 вариантов одновременно) задержках между ними,
- формирование децимированного потока данных для последующего поиска сигнала с помощью модулей FSE или программного обеспечения,

- формирование индивидуальных кодов спутниковых сигналов для сторонних устройств.

Остальные блоки выполняют вспомогательные функции и их внутренняя структура не слишком важна для понимания принципов работы TC7.

23.8.1 Многократное использование аппаратных ресурсов (time sharing)

При использовании современных полупроводниковых технологий предельная тактовая частота блоков обработки сигнала в несколько раз (десятки, иногда сотни) превосходит частоту, необходимую для обработки сигнала в реальном времени. Это позволяет организовать многократное использование аппаратных ресурсов с разделением по времени (time sharing) для последовательного решения нескольких однотипных задач, что, в свою очередь, позволяет обходиться меньшим количеством аппаратных ресурсов и уменьшает занимаемую на кристалле площадь. Слежение за множеством сигналов от различных спутников с помощью методов, осуществляющих многократные вычисления корреляции между принимаемым сигналом и опорной последовательностью, как раз относится к подобному классу задач. В Navicore5L принцип разделения по времени позволяет осуществлять слежение за десятками спутниковых сигналов всего четырьмя устройствами обработки (PU — Processing Unit).

Блок TC7 осуществляет обработку данных, предварительно сохраненных в памяти внешнего по отношению к нему устройства (Navicore5L). Контроллер прямого доступа к памяти DMA осуществляет подачу отсчетов обрабатываемого сигнала и данных конфигурации на входы TC7, сохранение результатов обработки, а также сохранение и восстановление конфигурации TC7 (контекста).

Работа с непрерывными сигналами, принимаемыми из эфира, организована путем разбиения сигналов на кадры (фреймы) фиксированной длительности по времени. В Navicore5L фреймы формируются модулями интерфейса с АЦП и преобразования частот DDC, объединенными оболочкой DDC_TOP. Каждый фрейм данных имеет заголовок, описывающий формат данных и их количество. Устройство DMA осуществляет запись блоков данных во внешнюю память через интерфейс AXI-master. После того, как в памяти оказались фрагменты последовательностей всех входных сигналов, соответствующие текущему интервалу времени, записанные блоки начинают передаваться с помощью DMA на обработку в TC7.

Обработка блоков данных в TC7 организована с помощью циклов. Во время одной итерации цикла осуществляется обработка блоков сигналов, относящихся к одному и тому же интервалу времени. Обработка каждого фрейма сигнала производится многократно с применением разных параметров – частот, задержек, дальномерных кодов. Кроме того, в целях сохранения непрерывности обработки, для каждого набора «сигнал + параметры обработки» требуется сохранение состояния модуля в конце обработки текущего блока и его восстановление перед обработкой следующего блока. Для этого в модуле TC7

реализован механизм автоматической загрузки параметров, загрузки и сохранения состояния.

Набор данных, содержащий настройки и состояние модуля, далее в документе называется «контекст». Каждый контекст описывает состояние от 1 до 4 «виртуальных каналов» обработки. Виртуальные каналы, описанные в одном контексте, активизируются путем загрузки состояний в 4 физических PU, после чего PU одновременно выполняют обработку одного и того же фрейма сигнала.

Контекст должен быть загружен непосредственно перед обработкой очередного фрейма данных и должен быть сохранен после обработки. Общее количество виртуальных каналов определяется количеством циклов загрузки контекста, обработки и сохранения контекста, приходящимся на один временной интервал. Количество циклов ограничено соотношением системной частоты и частоты дискретизации обрабатываемого сигнала.

В упрощенном виде цикл обработки данных четырехканальным модулем TC7 можно представить в следующем виде:

Старт обработки интервала времени №0
Загрузка контекста виртуальных каналов №0-3 (vc0-vc3) для сигнала №0 (sig0)
Обработка блока sig0
Сохранение контекста vc0-vc3
Загрузка контекста vc4-vc7 для sig0
Обработка блока sig0
Сохранение контекста vc4-vc7
Загрузка контекста vc8-vc11 для sig1
Обработка блока sig1
Сохранение контекста vc8-vc11
……
Сохранение результатов обработки из встроенных потоков TC7 во внешнюю память
Старт обработки интервала №1

И т.д.

Последовательность выполняется без участия микроконтроллера и задается цепочкой дескрипторов задач для машины DMA.

23.8.2 Внутренняя память

Внутренняя память состоит из ROM размером 64 Кбайт и RAM размером 32 Кбайт. Оба типа памяти доступны в едином пространстве адресов. Адресация, чтение и запись памяти осуществляется 64-битными словами при доступе со стороны внутренних блоков TC7 и через порты DMA. При доступе через регистры ICTL разрядность доступа — 32 бита.

ROM TC7 содержит информацию о 10 стандартных типах кодов — дескрипторы и базовые последовательности для генерации индивидуальных дальномерных кодов спутников. Они применяются автоматически при задании типов кода от 0 до 9 в поле контекста

PRN_MODE. Для кодов, отсутствующих в ROM, а также нестандартных или на текущий момент неизвестных кодов дескрипторы и базовые коды могут быть записаны в RAM

RAM может быть использована для нескольких назначений:

- хранение дескрипторов и базовых последовательностей нестандартных кодов,
- хранение буферов потокового вывода (результаты корреляции, децимации, генерации кода),
- хранение произвольных данных.

В следующей таблице даны начальные адреса всех объектов, расположенных в памяти TC7 и используемых по различным назначениям. Видно, что в RAM области хранения кодов и потоков вывода перекрываются между собой. Это позволяет перераспределять память в зависимости от задач.

Таблица 23.5. Распределение памяти TC7

Раздел	Начальный адрес			Описание
	Байты	32b слова	64b слова	
ROM	0x0	0x0	0x0	Начало ROM
RAM start	0x10000	0x4000	0x2000	Начало RAM
CDR_ADDR10	0x10000	0x4000	0x2000	Начало области дескрипторов кодов при установке поля контекста PRN_MODE==10
CDR_ADDR11	0x10400	0x4100	0x2080	...при PRN_MODE==11
CDR_ADDR12	0x10800	0x4200	0x2100	...при PRN_MODE==12
CDR_ADDR13	0x10C00	0x4300	0x2180	...при PRN_MODE==13
CDR_ADDR14	0x11000	0x4400	0x2200	...при PRN_MODE==14
CDR_ADDR15	0x11400	0x4500	0x2280	...при PRN_MODE==15
CDOUT_START_ADDR0	0x10000	0x4000	0x2000	Начальный адрес потока вывода #0
CDOUT_START_ADDR1	0x11000	0x4400	0x2200	Начальный адрес потока вывода #1
CDOUT_START_ADDR2	0x12000	0x4800	0x2400	Начальный адрес потока вывода #2
CDOUT_START_ADDR3	0x13000	0x4C00	0x2600	Начальный адрес потока вывода #3
CDOUT_START_ADDR4	0x14000	0x5000	0x2800	Начальный адрес потока вывода #4
CDOUT_START_ADDR5	0x15000	0x5400	0x2A00	Начальный адрес потока вывода #5
CDOUT_START_ADDR6	0x16000	0x5800	0x2C00	Начальный адрес потока вывода #6
CDOUT_START_ADDR7	0x17000	0x5C00	0x2E00	Начальный адрес потока вывода #7

Внутренняя память типа ROM содержит данные, предназначенные для формирования кодовых последовательностей.

Кроме возможности хранения произвольных данных, у внутренней памяти TC7 типа RAM есть два специальных назначения – содержание исходных данных для формирования кодов пользователя (для использования данной возможности необходимо обратиться в службу поддержки) и предоставление места для формирования потоков выходных данных TC7. Для этого в ней может быть выделено несколько областей с фиксированными начальными

адресами, которые перечислены в таблице «Распределение памяти TC7» в начале раздела. Адреса областей зависят от программирования кодов и потоков вывода, что позволяет использовать буферы разного размера.

23.9 Машина быстрого поиска FSE

Машина быстрого поиска FSE при каждом запуске осуществляет поиск наиболее вероятной гипотезы временной задержки сигнала для заданных гипотез частоты Доплера и кода сигнала. Проверка осуществляется с помощью согласованного фильтра с циклическим когерентным накоплением нескольких периодов кода и последующим некогерентным накоплением результатов нескольких когерентных накоплений. Результатом поиска является амплитуда максимального полученного корреляционного пика и его позиция в чипах относительно начала первого фрейма обрабатываемого сигнала. Также возможно получение полной выборки некогерентного накопления по всем временным позициям.

Обобщенная блок-схема FSE представлена на рисунке

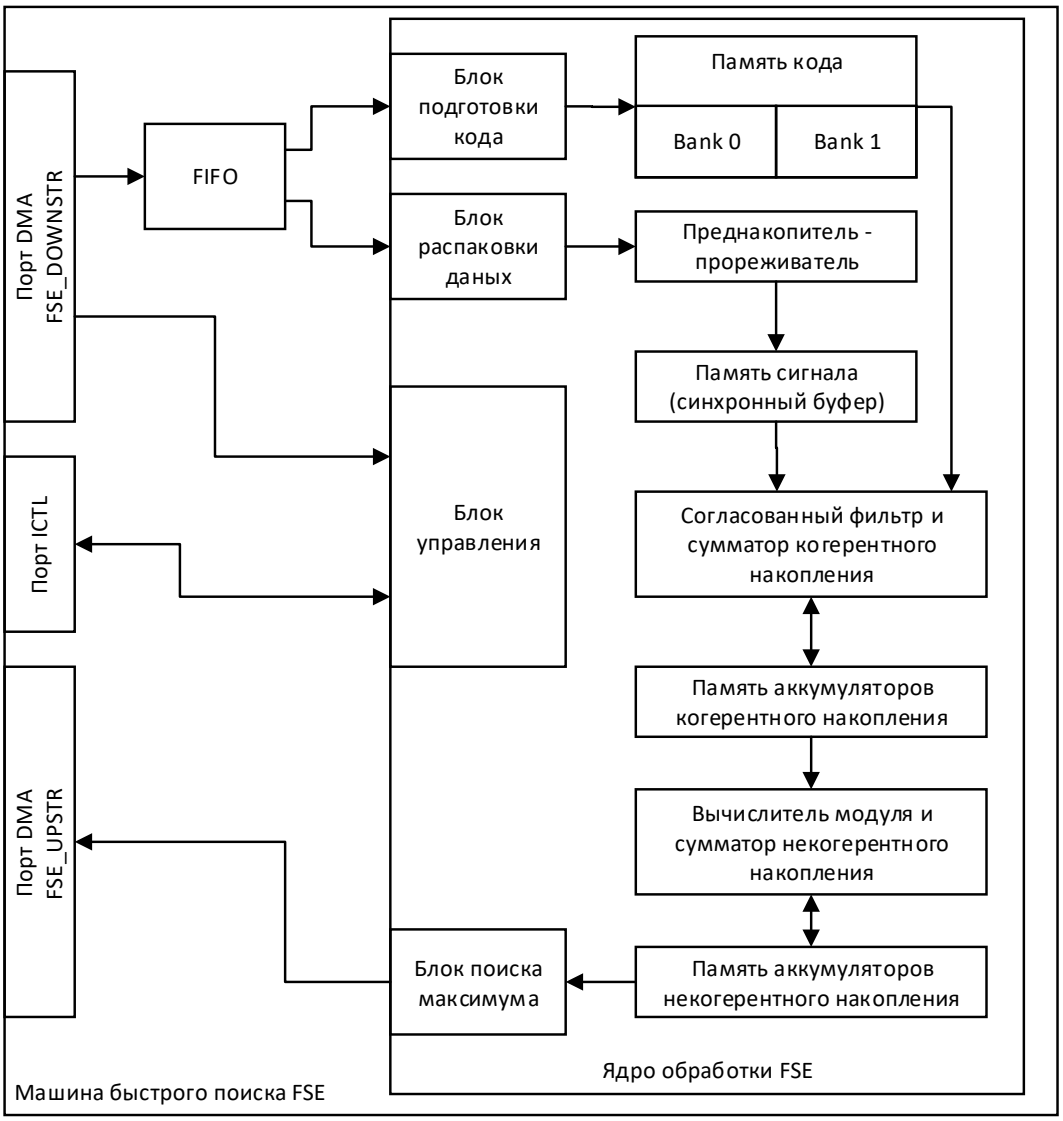


Рисунок 23.6. Блок-схема FSE

Машина может искать следующие виды сигналов существующих систем позиционирования:

- GPS L1/L2 CA
- ГЛОНАСС L1/L2 CT
- Beidou B2
- GALILEO E1

Возможен, также, поиск других видов сигналов, если длина их кода совпадает с длиной кодов перечисленных выше систем.

Всего машина поиска вычисляет оценку вероятности 2046 или 2044 гипотез задержки сигнала. Таким образом для сигналов GPS L1/L2 CA и ГЛОНАСС CT поиск осуществляется с шагом до 1/2 и 1/4 чипа соответственно. Потери мощности оценки, обусловленные случайным смещением гипотез по времени, составляют при этом не более 25% для GPS и не более 12.5% для ГЛОНАСС.

Для сигнала Beidou B2 шаг гипотез задержки составляет 1 чип, поскольку длина эпохи дальномерного кода сигнала равна 2046 чипам, что приводит к потере до 50% мощности сигнала в худшем случае, т.е. при сдвиге всех гипотез на 0.5 чипа. Это ведет к уменьшению вероятности правильного обнаружения сигнала. FSE позволяет скомпенсировать данный эффект путем повторного запуска машины поиска с другой начальной фазой генератора в поле записи конфигурации CHIP_SHIFT на той же выборке сигнала, чем достигается проверка гипотез, сдвинутых на 1/2 чипа. В этом случае, ценой увеличения времени поиска в 2 раза, достигается уровень потерь, не превышающий таковой для сигнала GPS CA.

Эпоха сигнала Galileo E1 имеет длительность 4 мс и 4092 чипа, а его чиповая частота совпадает с частотой сигнала GPS CA. Каждый чип модулирован поднесущей BOC1 частотой 1 МГц. Т.е. на протяжении эпохи содержится 8184 интервалов сигнала с различными логическими состояниями. Поскольку память FSE вмещает только 2046 отсчетов сигнала, поиск временной позиции кода производится в несколько этапов частичными накоплениями.

При поиске Galileo E1 входной сигнал прореживается до частоты 511.5 кГц встроенными средствами FSE. Таким образом, загружаемый в синхронный буфер сигнал содержит отсчеты, относящиеся к одной и той же полуволне BOC1 и к только четным либо к только нечетным чипам сигнала. Загружаемый в память код, в свою очередь, разделяется на 2 банка памяти таким образом, что в банке 0 оказываются четные чипы кода (0,2,4…), а в банке 1 — нечетные (1,3,5…). FSE 2 раза обрабатывает массив данных со всеми одинаковыми настройками, кроме выбора банка кода и момента старта прореживания сигнала. При втором запуске меняется банк кода, а начало прореженного сигнала сдвигается ровно на один чип. Поскольку между двумя вызовами накопление не сбрасывается, в аккумуляторах накапливается полная мощность сигнала для заданной начальной фазы.

Для поиска временной задержки сигнала с точностью до 1/2 полуволны BOC1 необходимо осуществить 8 пар циклов поиска с постепенным приращением начальной фазы чипа с шагом 1/4 чипа. Таким образом, увеличив время обработки сигнала в 16 раз, можно обеспечить поиск сигнала GALILEO E1 без увеличения потерь мощности относительно сигнала GPS CA.

Данные о времени обработки (количество проходов) и потерях мощности при поиске сведены в таблицу:

Таблица 23.6. Потери мощности при различных параметрах поиска

Система и сигнал	Количество проходов накопления	Потери мощности
GLONASS CT L1/L2	1	12.5 %
GPS CA L1/L2	1	25 %
Beidou B2	1	50 %
Beidou B2	2	25 %
Galileo E1	8	50 %
Galileo E1	16	25 %

24. КОНТРОЛЛЕР ДИНАМИЧЕСКОЙ ПАМЯТИ (DDRMC)

24.1 Общая информация

Контроллер динамической памяти имеет следующий функционал:

- Поддержка протоколов DDR3, DDR3L, DDR4, LPDDR3, LPDDR4[1];
- Скорости передачи не менее 1600Мбит/с;
- 32-разрядная шина данных;
- Адресация двух параллельно подключенных микросхем памяти с помощью сигнала chip select, в режиме DDR4 адресация до 4 блоков 3DS с помощью сигнала Chip ID, адресация до 64 Гбайт внешней памяти;
- Поддерка коррекции ошибок кодом Хэмминга;

24.2 Система тактирования

Контроллер требует наличия следующих синхросигналов:

1. ddr_clk – тактовая частота интерфейса DDR, равная половине частоты sdram_ck / sdram_ck_n;
2. aclk – синхросигналы каналов AXI;
3. pclk – синхросигнал интерфейса APB;
4. atpg_clk – тестовый синхросигнал.

24.3 Система сброса

Контроллера требует наличия следующих сигналов сброса:

1. ddr_reset_n – сброс логики uMCTL2;
2. pub_reset_n – сброс логики PUB;
3. ddr_bor_n – сброс PHY при включении питания;
4. aresetn – сброс логики, связанной с каналами AXI, независимо для каждого канала;

[1] Максимальный объем памяти LPDDR4, поддерживаемый К1892ВМ21Я, составляет 8 Гбайт на контроллер.

5. presetn – сброс конфигурационной части.

24.4 Интерфейс

Конфигурирование контроллера осуществляется с помощью стандартного 32-битного интерфейса APB (версии 3.0) для uMCTL2 (psel[0]) и PUBL(psel[1]).

Передача данных осуществляется через интерфейс AXI (11 портов). Разрядность интерфейса данных отличается для разных каналов и варьируется от 32 до 256 бит. Сигналы AxCACHE, AxPROT не поддерживаются.

Внешний интерфейс SDRAM содержит 48 выводов данных, 40 выводов адреса/управления, MEMRESET_L, ALERT_N, калибровочные сигналы ZN_SENSE и ZN. Назначение выводов функциям зависит от режима работы контроллера и типа внешней памяти (таблица 24.1).

24.4.1 Соответсвие AXI портов мастерам микросхемы

Номер AXI порта контроллера	Источник запросов
0	Все мастера SDR подсистемы кроме контроллеров PCIe
1	Контроллер PCIe0 и PCIe1
2	Процессор ISP
3	Процессок GPU
4	Процессор VPU
5	Процессор дисплея (DP)
6	Центральный процессор
7	Все мастера сервисной подсистемы
8	Все мастера HSP подсистемы
9	Все мастера LSP0 подсистемы
10	Все мастера LSP1 подсистемы

24.5 Внешние выводы

Помимо стандартных выводов (AXI, APB, SDRAM), сигналов сброса и тактирования, контроллер имеет дополнительные сигналы.

К дополнительным следует отнести:

- Прерывания: dwc_ddrphy_int_n, ecc_corrected_err_intr, ecc_corrected_err_intr_fault, ecc_uncorrected_err_intr, ecc_ubncorrected_err_intr_fault, ecc_ap_err_inyr, ecc_ap_err_intr_fault, dfi_alert_err_intr, derate_temp_limit_intr, derate_temp_limit_intr_fault;
- Данные hif_mrr_data;
- Управление режимами памяти ram_sleep;
- Тестовые выводы dwc_ddrphy_dto, scanmode, atpg_mode, atpg_clk, scan_in, scan_reset_n, bs_clock, bs_mode, bs_capture, bs_shift, bs_shift_in, bs_shift_out;

- Выводы питания VDD, VSS, VDDQ, VSSQ, VREF;
- Аналоговые калибровочные выводы: BP_ZN, BP_ZN_SENSE.

24.5.1 Интерфейс чтения регистров памяти

Стандарт DDR4 JEDEC разрешает чтение Multi Purpose Register (MPR) памяти. Контроллер позволяет выполнять чтение и отслеживать результат через вывод hif_mrr_data. Читать MPR следует по одному для каждого чипа памяти для избежания конфликтов на шине. Положение прочитанных данных в hif_mrr_data не учитывает возможное изменение последовательности подключения байт на печатной плате.

Для чтения MPR выполняется последовательность:

- Если разрешена повторная передача (**CRCPARCTRL0.crc_parity_retry_enable** = 1) следует запретить операции чтения и записи на интерфейсе DFI установив **DBG1.dis_dq** = 1 и отслеживая **DBGCAM.wr_data_pipeline_empty** и **DBGCAM.rd_data_pipeline_empty**, чтобы убедиться, что все исходящике команды на DFI завершены.
  Отследить значение **CRCPARSTAT.cmd_in_err_window** до тех пор, пока оно не станет равно 0, чтобы убедиться, что не было ошибок четности. Если программное управление разрешено (**CRCPARCTL1.alert_wait_for_sw**), также следует отслеживать значения **CRCPARSTAT.dfi_alert_err_int** и **CRCPARSTAT.dfi_alert_err_fatl_int** во время отслеживания **CRCPARSTAT.cmd_in_err_window**. Если один из указанных регистров установился до окончания отслеживания, процедура повторной пересылки должна быть заверешена до выполнения последующих шагов.
  Следует обратить внимание, что, если чтение MPR происходит во время программного вмешательства в процедуру пересылки, не следует отслеживать **CRCPARSTAT.cmd_in_err_window**.
- Выполнить команды Mode Register Set (MRS) к регистру MR3 SDRAM, чтобы перевести память в режим MPR. Установить **MRCTRL0.mpr_en** = 0. Также следует задать MR3[1:0] и MR3[12:11].
- Дождаться **MRSTAT.mr_wr_busy** = 0. Записать **MRCTRL1.mr_data**, где **MRCTRL1.mr_data**[1:0] = 00.
- Записать **MTRCTRL0**. **MRCTRL0.mr_addr** = MPR Location, **MRCTRL0.mr_type** = read, **MRCTRL0.mr_wr** = 1, **MRCTRL0.mpr_en** = 1.
  Указанная последовательность приведет к запуску чтения MPR. Результат чтения отражается на выводе hif_mrr_data.
- Выйти из режима MPR выполнив команду MRS. **MRCTRL0.mpr_en** = 0.
- Разрешить чтение и запись на DFI записью **DBG1.dis_dq** = 0, если они ранее были отключены

В режиме MPR поддерживаются только операции с BL8 (Burst Length = 8). BC4 (Burst Chop 4) не поддерживается.

24.6 Регистры блока UMCTL2_REGS

Карта регистров UMCTL2_REGS приведена в Таблица 24.2.

Таблица 24.2. Карта регистров UMCTL2_REGS

№	Регистр	Описание	Смещение
1	MSTR	Основной регистр 0 (Master Register 0)	0x0
2	STAT	Регистр статуса режима работы (Operating Mode Status Register)	0x4
3	MRCTRL0	Регистры режима: регистр управления чтением/записью 0 (Mode Register Read/Write Control Register 0)	0x10
4	MRCTRL1	Регистры режима: регистр управления чтением/записью 1 (Mode Register Read/Write Control Register 1)	0x14
5	MRSTAT	Регистры режима: регистр управления чтением/записью (Mode Register Read/Write Status Register)	0x18
6	MRCTRL2	Регистры режима: регистр управления чтением/записью 2 (Mode Register Read/Write Control Register 2)	0x1c
7	DERATEEN	Регистр подключения перерасчёта температуры (Temperature Derate Enable Register)	0x20
8	DERATEINT	Регистр интервала перерасчёта температуры (Temperature Derate Interval Register)	0x24
9	DERATECTL	Регистр управления перерасчётом температуры (Temperature Derate Control Register)	0x2c
10	PWRCTL	Регистр управления низким питанием (Low Power Control Register)	0x30
11	PWRTMG	Регистр задержек низкого питания (Low Power Timing Register)	0x34
12	HWLPCTL	Регистр аппаратного управления низким питанием (Hardware Low Power Control Register)	0x38
13	RFSHCTL0	Регистр управления регенерацией 0 (Refresh Control Register 0)	0x50
14	RFSHCTL1	Регистр управления регенерацией 1 (Refresh Control Register 1)	0x54
15	RFSHCTL3	Регистр управления регенерацией 3 (Refresh Control Register 3)	0x60
16	RFSHTMG	Регистр задержек регенерации (Refresh Timing Register)	0x64
17	RFSHTMG1	Регистр задержек регенерации 1 (Refresh Timing Register 1)	0x68
18	ECCCFG0	Регистр конфигурации ECC 0 (ECC Configuration Register 0)	0x70

№	Регистр	Описание	Смещение
19	ECCCFG1	Регистр конфигурации ECC 1 (ECC Configuration Register 1)	0x74
20	ECCSTAT	Регистр состояния SECDED ECC (SECDED ECC Status Register)	0x78
21	ECCCTL	Регистр очистки ECC (ECC Clear Register)	0x7C
22	ECCERRCNT	Регистр счетчика ошибок (ECC Error Counter Register)	0x80
23	ECCCADDR0	Регистр адреса корректируемой ошибки 0 (ECC Corrected Error Address Register 0)	0x84
24	ECCCADDR1	Регистр адреса корректируемой ошибки 1 (ECC Corrected Error Address Register 1)	0x88
25	ECCCSYN0	Регистр синдрома корректируемой ошибки 0 (ECC Corrected Syndrome Register 0)	0x8c
26	ECCCSYN1	Регистр синдрома корректируемой ошибки 1 (ECC Corrected Syndrome Register 1)	0x90
27	ECCCSYN2	Регистр синдрома корректируемой ошибки 2 (ECC Corrected Syndrome Register 2)	0x94
28	ECCBITMASK 0	Регистр маски корректируемых бит 0 (ECC Corrected Data Bit Mask 0)	0x98
29	ECCBITMASK 1	Регистр маски корректируемых бит 1 (ECC Corrected Data Bit Mask 1)	0x9c
30	ECCBITMASK 2	Регистр маски корректируемых бит 2 (ECC Corrected Data Bit Mask 2)	0xa0
31	ECCUADDR0	Регистр адреса некорректируемой ошибки 0 (ECC Uncorrected Error Address Register 0)	0xa4
32	ECCUADDR1	Регистр адреса некоректируемой ошибки 1 (ECC Uncorrected Error Address Register 1)	0xa8
33	ECCUSYN0	Регистр синдрома некорректируемой ошибки 0 (ECC Uncorrected Syndrome Register 0)	0xac
34	ECCUSYN1	Регистр синдрома некорректируемой ошибки 1 (ECC Uncorrected Syndrome Register 1)	0xb0
35	ECCUSYN2	Регистр снидрома некорректируемой ошибки 2 (ECC Uncorrected Syndrome Register 2)	0xb4
36	ECCPOISONA DDR0	Регистр адреса внесенной ошибки 0 (ECC Data Poisoning Address Register 0)	0xb8
37	ECCPOISONA DDR1	Регистр адреса внесенной ошибки 1 (ECC Data Poisoning Address Register 1)	0xbc
38	CRCPARCTL0	Регистр управления контролем четности и CRC 0 (CRC Parity Control Register 0)	0xc0
39	CRCPARCTL1	Регистр управления контролем четности и CRC 1 (CRC Parity Control Register1)	0xc4
40	CRCPARSTAT	Регистр статуса контроля четности и CRC (CRC Parity Status Register)	0xcc
41	INIT0	Регистр инициализации SDRAM 0 (SDRAM Initialization Register 0)	0xd0

№	Регистр	Описание	Смещение
42	INIT1	Регистр инициализации SDRAM 1 (SDRAM Initialization Register 1)	0xd4
43	INIT2	Регистр инициализации SDRAM 2 (SDRAM Initialization Register 2)	0xd8
44	INIT3	Регистр инициализации SDRAM 3 (SDRAM Initialization Register 3)	0xdc
45	INIT4	Регистр инициализации SDRAM 4 (SDRAM Initialization Register 4)	0xe0
46	INIT5	Регистр инициализации SDRAM 5 (SDRAM Initialization Register 5)	0xe4
47	INIT6	Регистр инициализации SDRAM 6 (SDRAM Initialization Register 6)	0xe8
48	INIT7	Регистр инициализации SDRAM 7 (SDRAM Initialization Register 7)	0xec
49	DIMMCTL	Регистр управления DIMM (DIMM Control Register)	0xf0
50	RANKCTL	Регистр управления рангом (Rank Control Register)	0xf4
51	DRAMTMG0	Регистр задержек SDRAM 0 (SDRAM Timing Register 0)	0x100
52	DRAMTMG1	Регистр задержек SDRAM 1 (SDRAM Timing Register 1)	0x104
53	DRAMTMG2	Регистр задержек SDRAM 2 (SDRAM Timing Register 2)	0x108
54	DRAMTMG3	Регистр задержек SDRAM 3 (SDRAM Timing Register 3)	0x010C
55	DRAMTMG4	Регистр задержек SDRAM 4 (SDRAM Timing Register 4)	0x0110
56	DRAMTMG5	Регистр задержек SDRAM 5 (SDRAM Timing Register 5)	0x0114
57	DRAMTMG6	Регистр задержек SDRAM 6 (SDRAM Timing Register 6)	0x0118
58	DRAMTMG7	Регистр задержек SDRAM 7 (SDRAM Timing Register 7)	0x11c
59	DRAMTMG8	Регистр задержек SDRAM 8 (SDRAM Timing Register 8)	0x120
60	DRAMTMG9	Регистр задержек SDRAM 9 (SDRAM Timing Register 9)	0x124
61	DRAMTMG11	Регистр задержек SDRAM 11 (SDRAM Timing Register 11)	0x12c
62	DRAMTMG12	Регистр задержек SDRAM 12 (SDRAM Timing Register 12)	0x130
63	DRAMTMG13	Регистр задержек SDRAM 13 (SDRAM Timing Register 13)	0x134
64	DRAMTMG14	Регистр задержек SDRAM 14 (SDRAM Timing Register 14)	0x138

№	Регистр	Описание	Смещение
65	DRAMTMG15	Регистр задержек SDRAM 15 (SDRAM Timing Register 15)	0x13C
66	ZQCTL0	Регистр управления ZQ 0 (ZQ Control Register 0)	0x180
67	ZQCTL1	Регистр управления ZQ 1 (ZQ Control Register 1)	0x184
68	ZQCTL2	Регистр управления ZQ 2 (ZQ Control Register 2)	0x188
69	ZQSTAT	Регистр статуса ZQ (ZQ Status Register)	0x18C
70	DFITMG0	Регистр задержек DFI 0 (DFI Timing Register 0)	0x190
71	DFITMG1	Регистр задержек DFI 1 (DFI Timing Register 1)	0x194
72	DFILPCFG0	Регистр конфигурирования интерфейса низкого питания DFI 0 (DFI Low Power Configuration Register 0)	0x198
73	DFILPCFG1	Регистр конфигурирования интерфейса низкого питания DFI 1 (DFI Low Power Configuration Register 1)	0x19C
74	DFIUPD0	Регистр обновлений DFI 0 (DFI Update Register 0)	0x1a0
75	DFIUPD1	Регистр обновлений DFI 1 (DFI Update Register 1)	0x1a4
76	DFIUPD2	Регистр обновлений DFI 2 (DFI Update Register 2)	0x1a8
77	DFIMISC	Регистр прочих настроек DFI (DFI Miscellaneous Control Register)	0x1b0
78	DFITMG2	Регистр задержек DFI 2 (DFI Timing Register 2)	0x1b4
79	DFISTAT	Регистр статуса DFI (DFI Status Register)	0x1bC
80	DBICTL	Регистр управления DM/DBI (DM/DBI Control Register)	0x1c0
81	DFIPHYMSTR	Регистр интерфейса управления PHY в DFI (DFI PHY Master)	0x1c4
82	ADDRMAP0	Регистр соответствия адресов 0 (Address Map Register 0)	0x200
83	ADDRMAP1	Регистр соответствия адресов 1 (Address Map Register 1)	0x204
84	ADDRMAP2	Регистр соответствия адресов 2 (Address Map Register 2)	0x208
85	ADDRMAP3	Регистр соответствия адресов 3 (Address Map Register 3)	0x20c
86	ADDRMAP4	Регистр соответствия адресов 4 (Address Map Register 4)	0x210
87	ADDRMAP5	Регистр соответствия адресов 5 (Address Map Register 5)	0x214

№	Регистр	Описание	Смещение
88	ADDRMAP6	Регистр соответствия адресов 6 (Address Map Register 6)	0x218
89	ADDRMAP7	Регистр соответствия адресов 7 (Address Map Register 7)	0x21c
90	ADDRMAP8	Регистр соответствия адресов 8 (Address Map Register 8)	0x220
91	ADDRMAP9	Регистр соответствия адресов 9 (Address Map Register 9)	0x224
92	ADDRMAP10	Регистр соответствия адресов 10 (Address Map Register 10)	0x228
93	ADDRMAP11	Регистр соответствия адресов 11 (Address Map Register 11)	0x22c
94	ODTCFG	Регистр конфигурирования ODT (ODT Configuration Register)	0x240
95	ODTMAP	Регистр соответствия терминации ODT/ранга (ODT/Rank Map Register)	0x244
96	SCHED	Регистр управления планировщиком (Scheduler Control Register)	0x250
97	SCHED1	Регистр управления планировщиком 1 (Scheduler Control Register 1)	0x254
98	PERFHPR1	Регистр CAM чтений с высоким приоритетом 1 (High Priority Read CAM Register 1)	0x25c
99	PERFLPR1	Регистр CAM чтений с низким приоритетом 1 (Low Priority Read CAM Register 1)	0x264
100	PERFWR1	Регистр записей CAM 1 (Write CAM Register 1)	0x26c
101	DBG0	Регистр отладки 0 (Debug Register 0)	0x300
102	DBG1	Регистр отладки 1 (Debug Register 1)	0x304
103	DBGCAM	Регистр отладки CAM (CAM Debug Register)	0x308
104	DBGCMD	Регистр отладки команд (Command Debug Register)	0x30c
105	DBGSTAT	Регистр статуса отладки (Status Debug Register)	0x310
106	DBGCAM1	Регистр отладки CAM 1 (CAM Debug Register 1)	0x318
107	SWCTL	Подключение настроек программирования регистров через программное обеспечение (Software Register Programming Control Enable)	0x320
108	SWSTAT	Статус настроек программирования регистров через программное обеспечения (Software Register Programming Control Status)	0x324
109	POISONCFG	Регистр конфигурации сбоев AXI (AXI Poison Configuration Register)	0x36c
110	POISONSTAT	Регистр статуса сбоев AXI (AXI Poison Status Register)	0x370

№	Регистр	Описание	Смещение
111	ADVECCINDEX	Расширенный регистр индексов ECC (Advanced ECC Index Register)	0x374
112	ECCPOISONPAT0	Регистр шаблона внесенной ошибки 0 (ECC Poison Pattern 0 Register)	0x37c
113	ECCPOISONPAT2	Регистр шаблона внесенной ошибки 2 (ECC Poison Pattern 2 Register)	0x384
114	ECCAPSTAT	Регистр статуса защиты адреса ECC (Address protection within ECC Status Register)	0x388
115	DERATESTAT	Регистр статуса снижения максимально допустимых значений температуры (Temperature Derate Status Register)	0x3f0

24.6.1 Основной регистр 0 (MSTR)

Поля регистра MSTR описаны в Таблица 24.3.

Таблица 24.3. Основной регистр 0

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	ddr3	Этот бит управляет выбором DDR3 SDRAM: 1 - используется устройство DDR3 SDRAM; 0 - используется устройство, отличное от DDR3. Присутствует только в реализациях, поддерживающих DDR3. Режим программирования: статический.	0x1	R/W
[1]	–	Зарезервировано.	0	
[2]	lpddr2	Этот бит управляет выбором LPDDR2 SDRAM: 1 - используется устройство LPDDR2 SDRAM; 0 - используется устройство, отличное от LPDDR2. Присутствует только в реализациях, сконфигурированных для поддержки LPDDR2. Режим программирования: статический.	0x0	R/W
[3]	lpddr3	Этот бит управляет выбором LPDDR3 SDRAM: 1 - используется устройство LPDDR3 SDRAM; 0 - используется устройство, отличное от LPDDR3. Присутствует только в реализациях, сконфигурированных для поддержки LPDDR3. Режим программирования: статический.	0x0	R/W
[4]	ddr4	Этот бит управляет выбором DDR4 SDRAM: 1 - используется устройство DDR4 SDRAM; 0 - используется устройство, отличное от DDR4. Присутствует только в реализациях, сконфигурированных для поддержки DDR4. Режим программирования: статический.	0x0	R/W
[5]	lpddr4	Этот бит управляет выбором LPDDR4 SDRAM:	0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		1 - используется устройство LPDDR4 SDRAM; 0 - используется устройство, отличное от LPDDR4. Присутствует только в реализациях, сконфигурированных для поддержки LPDDR4. Режим программирования: статический.		
[8:6]	–	Зарезервировано.	0	
[9]	burstchop	Когда установлен, подключает режим разделения пакета (burst-chop) (режим BC4 или 8 "на лету") в DDR3/DDR4. Для команд чтения проверка разделения пакета выполняется только в HIF-конфигурациях (UMCTL2_INCL_ARB не установлен), при условиях, что подключен режим полной ширины шины (MSTR.data_bus_width = 00) и MEMC_BURST_LENGTH=8 или 16. Для команд записи проверка разделения пакета выполняется только если подключена возможность частичной записи (UMCTL2_PARTIAL_WR=1) и CRC отключен (CRCPARCTL1.crc_enable = 0). Режим BC4 (фиксированный) не поддерживается. Режим программирования: статический.	0x0	R/W
[11:10]	–	Зарезервировано.	0x0	
[13:12]	data_bus_width	Это поле позволяет выбрать пропорцию ширин шины DQ, используемую в SDRAM. 00 - полная ширина шины DQ к SDRAM; 01 - половина ширины шины DQ к SDRAM; 10 - четверть ширины шины DQ к SDRAM; 11 - зарезервировано. Следует отметить, что режим полуширины шины поддерживается только когда ширина шины SDRAM кратна 16, а режим четверти ширины шины поддерживается, когда ширина шины SDRAM кратна 32 и установлен параметр конфигурации MEMC_QBUS_SUPPORT. Ширина шины относится к ширине шины DQ (исключая любую ширину ECC). Режим программирования: статический.	0x0	R/W
[14]	–	Зарезервировано.	0	
[15]	dll_off_mode	В это поле следует устанавливать 1, когда uMCTL2 и DRAM нужно поместить в режим с отключенным DLL для низкочастотной работы. В это поле следует устанавливать 0, чтобы поместить uMCTL2 и DRAM в режим с подключенным DLL для работы в нормальном диапазоне частот. Если подключен контроль чётности/CRC в DDR4 (CRCPARCTL1.crc_parity_retry_enable = 1), dll_off_mode не поддерживается, и в этот бит необходимо установить 0. Режим программирования: квазидинамическая группа 2.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
[19:16]	burst_rdwr	Используемая в SDRAM длина импульса: 0001 - длина импульса 2 (поддерживается только для mDDR); 0010 - длина импульса 4; 0100 - длина импульса 8; 1000 - длина импульса 16 (поддерживается только для mDDR, LPDDR2, и LPDDR4); Все остальные значения зарезервированы. Это поле управляет размером импульса, используемым для доступа к SDRAM. Оно должно соответствовать установке длины импульса в регистре режима в SDRAM (для динамического ("на лету") режима BC4/8 для DDR3 и DDR4, следует устанавливать в это поле 0x0100). Длина импульса 2 не поддерживается портами AXI когда длина MEMC_BURST_LENGTH=8. Длина импульса 2 поддерживается только когда контроллер работает в режиме соотношения частот 1:1. Для DDR3, DDR4 и LPDDR3, в это поле необходимо устанавливать 0x0100 (BL8). Для LPDDR4, в это поле необходимо устанавливать 0x1000 (BL16). Режим программирования: статический.	0x4	R/W
[23:20]	–	Зарезервировано.	0x0	
[25:24]	active_ranks	Присутствует только для мультиранговых конфигураций. Каждый бит соответствует одному рангу. Для двухранговых конфигураций присутствуют только биты [25:24]. 1: загружен (populated); 0: незагружен (unpopulated). LSB - наименьший номер ранга. Для двух рангов допустимы следующие комбинации: 01: один ранг; 11: два ранга; Остальные значения: зарезервировано. Для четырех рангов допустимы следующие комбинации: 0001: один ранг; 0011: два ранга; 1111: четыре ранга. Режим программирования: статический.	0x3	R/W
[29:26]	–	Зарезервировано.	0x0	
[31:30]	device_config	Указывает конфигурацию устройства, используемого в системе. 00: устройство x4; 01: устройство x8; 10: устройство x16; 11: устройство x32; Режим программирования: статический.	0x0	R/W

24.6.2 Регистр статуса режима работы (STAT)

Поля регистра STAT описаны в Таблица 24.4.

Таблица 24.4. Регистр статуса режима работы

Диапазон	Название	Описание	Исходное значение	Тип доступа
[2:0]	operating_mode	Режим работы. Это поле имеет ширину 3 бита в конфигурациях с поддержкой mDDR/LPDDR2/LPDDR3/LPDDR4/DDR4, и ширину 2 бита - во всех остальных конфигурациях. Значения битов для реализаций, не включающих модули mDDR/LPDDR2/LPDDR3/LPDDR4 и модули DDR4: 00 - инициализация; 01 - стандартный режим; 10 - режим пониженного питания; 11 - режим саморегенерации. Значения битов для реализаций mDDR/LPDDR2/LPDDR3 или DDR4: 000 - инициализация; 001 - стандартный режим; 010 - режим пониженного питания; 011 - режим саморегенерации. 1XX - режим глубоко пониженного питания / максимального энергосбережения (Deep Power Down / Maximum Power Saving Mode). Значения битов для реализаций LPDDR4: 000 - инициализация; 001 - стандартный режим; 010 - режим пониженного питания; 011 - режим саморегенерации / саморегенерации при пониженном питании. Режим программирования: статический. Условия переменного бита [x]: "(MEMC_MOBILE_OR_LPDDR2_OR_DDR4_EN==1) ? 3 : 2" - 1.	0x0	R
[3]	–	Зарезервировано.	0	
[5:4]	selfref_type	Указывает, осуществлен ли вход в режим саморегенерации (кроме LPDDR4), или вход в режим саморегенерации при пониженном питании (LPDDR4). Также указывает, осуществляется ли этот вход только за счёт автоматической регенерации (Automatic Self Refresh) или нет. 00 - SDRAM не находится в режиме саморегенерации (кроме LPDDR4) или саморегенерации при пониженном питании (LPDDR4). Если повторная попытка разрешена параметром CRCPARCTRL1.crc_parity_retry_enable, это также указывает, что команда входа в режим саморегенерации (Self-Refresh Entry command) всё	0x0	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
		еще находится в окне ошибки четности (parity error), или повтор попытки всё еще в процессе выполнения. 11 - SDRAM находится в режиме саморегенерации (кроме LPDDR4) или саморегенерации при пониженном питании (LPDDR4), вход в который спровоцирован строго автоматической регенерацией. Если повторная попытка разрешена, это также гарантирует, что команда входа в режим саморегенерации выполняется правильно, без ошибки четности. 10 - SDRAM находится в режиме саморегенерации (кроме LPDDR4) или саморегенерации при пониженном питании (LPDDR4). Вход произведен не только за счет автоматической регенерации. Он может быть спровоцирован аппаратным интерфейсом низкого питания и/или программно (PWRCTL.selfref_sw). Если повторная попытка разрешена, это также гарантирует, что команда входа в режим саморегенерации выполняется правильно, без ошибки четности. 01 - SDRAM находится в режиме саморегенерации, который спровоцирован запросом от главного модуля PHY. Режим программирования: статический.		
[7:6]	–	Зарезервировано.	0	
[9:8]	selfref_state	Режим саморегенерации. Это поле указывает состояние саморегенерации или саморегенерации при низком питании для LPDDR4. Это регистр используется для смены частот и доступа на запись/чтение от регистра режима (MRR/MRW) в процессе саморегенерации. 00 - SDRAM не находится в режиме саморегенерации; 01 - саморегенерация 1; 10 - саморегенерация при низком питании; 11 - саморегенерация 2. Режим программирования: статический.	0x0	R
[11:10]	–	Зарезервировано.	0	
[12]	selfref_cam_not_empty	Саморегенерация в условиях, когда ячейки ассоциативной памяти (CAMs) не пусты. Это поле устаналивается в значение "1", когда вход в режим саморегенерации произведен, но ячейки ассоциативной памяти не очищены. Обнуляется после выхода из саморегенерации. Режим программирования: статический.	0x0	R
[31:13]	–	Зарезервировано.	0	

24.6.3 Регистры режима: регистр управления чтением/записью 0 (MRCTRL0)

Замечание: не следует подключать одновременно более чем одно из следующих полей:

- sw_init_int;

- pda_en;

- mpr_en.

Поля регистра MRCTRL0 описаны в Таблица 24.5.

Таблица 24.5. Регистры режима: регистр управления чтением/записью 0

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	mr_type	Указывает операцию регистра режима: чтение или запись. Используется только для LPDDR2/LPDDR3/LPDDR4/DDR4. 0 - запись; 1 - чтение. Режим программирования: динамический.	0x0	R/W
[1]	mpr_en	Указывает операцию регистра режима: MRS или WR/RD для MPR (поддерживается только для DDR4). 0 - MRS; 1 - WR/RD для MPR (многоцелевого регистра). Режим программирования: динамический.	0x0	R/W
[2]	pda_en	Указывает операцию регистра режима MRS в режиме PDA или без него. 0 - MRS; 1 - MRS в режиме адресуемости по DRAM (Per DRAM Addressability). Следует отметить, что, когда pba_mode=1, инициируется доступ PBA вместо доступа PDA. Режим программирования: динамический.	0x0	R/W
[3]	sw_init_int	Указывает, разрешено ли вмешательство программного обеспечения через MRCTRL0/MRCTRL1 перед запуском автоматического алгоритма инициализации SDRAM. Для DDR4 этот бит может быть использован для инициализации DDR4 RCD (MR7) перед автоматической инициализацией SDRAM. Для LPDDR4 этот бит может быть использован для программирования добавочных регистров режима перед автоматической инициализацией SDRAM, если это необходимо. В двухканальном режиме LPDDR4 (LPDDR4 dual channel mode) такое программирование должно быть произведено предварительно для обоих каналов. Следует отметить, что этот бит должен быть обнулен после завершения работы программного обеспечения (Software operation). В противном случае, алгоритм инициализации SDRAM не перезапустится.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		0 - программное вмешательство не разрешено; 1 - программное вмешательство разрешено. Режим программирования: динамический.		
[11:4]	mr_rank	Управляет тем, к какому рангу обращается MRCTRL0.mr_wr. В стандартном случае желательно произвести обращение ко всем рангам, поэтому во все биты должна быть установлена «1». Однако, для мульти-ранговых UDIMMs/RDIMMs/LRDIMMs, которые реализуют дублирование адресов (address mirroring), может возникнуть необходимость индивидуальных обращений к рангам. Примеры (в предположении, что uMCTL2 сконфигурирована для 4 рангов): 0x1 - выбран только ранг 0; 0x2 - выбран только ранг 1; 0x5 - выбраны ранги 0 и 2; 0xA - выбраны ранги 1 и 3; 0xF - выбраны ранги 0, 1, 2 и 3. Режим программирования: динамический.	0x3	R/W
[15:12]	mr_addr	Адрес регистра режима, который должен быть записан в: 0000 - MR0; 0001 - MR1; 0010 - MR2; 0011 - MR3; 0100 - MR4; 0101 - MR5; 0110 - MR6; 0111 - MR7. Не относится к LPDDR2/LPDDR3/LPDDR4 (см. MRCTRL1.mr_data для адресации регистра режима в LPDDR2/LPDDR3/LPDDR4). Этот сигнал также используется для записи в управляющие слова (control words) чипа регистра в RDIMMs/LRDIMMs. В этом случае он соответствует битам адреса банка, посланным к RDIMM/LRDIMM. В случае работы с DDR4, бит [3:2] соответствует битам группы банка. Таким образом, в бит [3], так же, как и в бит [2:0], должно быть установлено соответствующее значение, которое считается одновременно значением дублирования адресов (Address Mirroring) для UDIMMs/RDIMMs/LRDIMMs, и значением инверсии вывода (Output Inversion) для RDIMMs/LRDIMMs. Режим программирования: динамический.	0x0	R/W
[29:16]	–	Зарезервировано.	0x0	
[30]	pba_mode	Указывает, выполнен ли доступ PBA (Per Buffer Addressability, адресуемость по буферу). При установке 1 в этот бит одновременно с установкой 1 в pda_en, uMCTL2 инициирует доступ PBA вместо доступа PDA. 0 - для режима адресуемости по DRAM (DRAM Addressability mode);	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		1 - для режима адресуемости по буферу (Buffer Addressability mode); Завершение доступа PBA подтверждается через MRSTAT.pda_done тем же способом, что и для PDA. Режим программирования: динамический.		
[31]	mr_wr	Установка 1 в этот бит регистра запускает операцию чтения или записи для регистра режима. Когда эта операция регистра режима MR завершена, uMCTL2 автоматически обнуляет этот бит. Запись в остальные поля этого регистра должна производиться только через отдельную транзакцию APB, перед тем, как устанавливать этот бит (mr_wr). Рекомендуется НЕ устанавливать этот сигнал, если работа ведется в режимах инициализации (Init), глубокого понижения питания (Deep power-down) или максимального энергосбережения (MPSM). Контролепригодность: readOnly. Режим программирования: динамический.	0x0	R/W

24.6.4 Регистры режима: регистр управления чтением/записью 1 (MRCTRL1)

Поля регистра MRCTRL1 описаны в Таблица 24.6.

Таблица 24.6. Регистры режима: регистр управления чтением/записью 1

Диапазон	Название	Описание	Исходное значение	Тип доступа
[17:0]	mr_data	Данные записи регистра режима для всех режимов, не относящихся к LPDDR2/LPDDR3/LPDDR4. Для LPDDR2/LPDDR3/LPDDR4, биты MRCTRL1[15:0] интерпретируются как адреса регистра режима [15:8] (MR Address) и данные регистра режима [7:0] (MR data) для записей, вне зависимости от чтений. Это поле имеет ширину 18 бит в конфигурациях с поддержкой DDR4 и 16 бит во всех остальных конфигурациях. Режим программирования: динамический.	0x0	R/W
[31:18]	–	Зарезервировано.	0	-

24.6.5 Регистры режима: регистр управления чтением/записью (MRSTAT)

Поля регистра MRSTAT описаны в Таблица 24.7.

Таблица 24.7. Регистры режима: регистр управления чтением/записью

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	mr_wr_busy	Система на кристалле (SoC) может инициировать операцию записи в регистр режима MR только если этот сигнал имеет низкое значение. Этот сигнал получает высокое значение в синхросигнале после того, как uMCTL2 принимает запрос на запись/чтение регистра режима MRW/MRR. Он получает низкое значение, когда команда MRW/MRR отправляется к SDRAM. Рекомендуется не выполнять MRW/MRR команд, когда 'MRSTAT.mr_wr_busy' имеет высокое значение. 0 - указывает, что система SoC может инициировать операцию записи в регистр режима. 1 - указывает, что операция записи в регистр режима находится в процессе выполнения. Режим программирования: динамический.	0x0	R
[7:1]	–	Зарезервировано	0	
[8]	pda_done	Система на кристалле (SoC) может инициировать операцию записи в регистр режима в режиме PDA/PBA только если этот сигнал имеет низкое значение. Этот сигнал получает высокое значение, когда к SDRAM происходит отправка трех идущих подряд команд MRS, относящихся к режиму PDA/PBA. Этот сигнал получает низкое значение, когда MRCTRL0.pda_en обнуляется. Таким образом, рекомендуется писать «0» в MRCTRL0.pda_en после того, как этот сигнал получает высокое значение, для того, чтобы подготовиться к выполнению операции PDA в следующий раз. 0 - указывает, что операция записи в регистр режима, связанная с PDA/PBA, находится в процессе выполнения или еще не началась. 1 - указывает, что операция записи в регистр режима, связанная с PDA/PBA, завершена. Режим программирования: динамический.	0x0	R
[31:9]	–	Зарезервировано	0	

24.6.6 Регистры режима: регистр управления чтением/записью 2 (MRCTRL2)

Поля регистра MRCTRL2 описаны в Таблица 24.8.

Таблица 24.8. Регистры режима: регистр управления чтением/записью 2

Диапазон	Название	Описание	Исходное значение	Тип доступа
[31:0]	mr_device_sel	Указывает устройство, которое следует выбрать во время команды статуса регистра режима (MRS), происходящей в режиме PDA. Каждый бит соответствует одному устройству. Например, bit[0] соответствует устройству "Device 0", бит [1] - "Device 1" и т. д. Следует запрограммировать значение "1", чтобы указать, что команда MRS должна быть применена к этому устройству.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Следует запрограммировать значение "0", чтобы указать, что команды MRS должны быть пропущены для этого устройства. Режим программирования: динамический.		

24.6.7 Регистр подключения перерасчёта температуры (DERATEEN)

Поля регистра DERATEEN описаны в Таблица 24.9.

Таблица 24.9. Регистр подключения перерасчёта температуры

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	derate_enable	Подключает перерасчёт: 0 - перерасчёт параметра задержки отключен; 1 - перерасчёт параметра задержки подключен с использованием значения чтения MR4. Присутствует только в реализациях, сконфигурированных для поддержки LPDDR2/LPDDR3/LPDDR4. Для режима, не относящегося к режимам LPDDR2/LPDDR3/LPDDR4, в это поле следует устанавливать «0». Режим программирования: динамический.	0x0	R/W
[2:1]	derate_value	Значение перерасчёта. 0 - перерасчет с использованием +1; 1 - перерасчет с использованием +2. Присутствует только в реализациях, сконфигурированных для поддержки LPDDR2/LPDDR3/LPDDR4. Должно иметь значение 0 для всех скоростных режимов LPDDR2, поскольку ограничивающая величина +1.875 нс меньше, чем тактовый период core_ddrc_core_clk. Для LPDDR3/4, если период core_ddrc_core_clk меньше, чем 1.875 нс, в это поле регистра должно быть установлено значение «1», в противном случае – «0». Режим программирования: квазидинамические группы 2 и 4.	0x0	R/W
[3]	–	Зарезервировано.	0	
[7:4]	derate_byte	Байт перерасчёта. Присутствует только в реализациях, сконфигурированных для поддержки LPDDR2/LPDDR3/LPDDR4. Указывает, который байт данных команды MRR используется для перерасчёта. Максимальное валидное значение зависит от MEMC_DRAM_TOTAL_DATA_WIDTH. Режим программирования: статический.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
[10:8]	rc_derate_value	Значение перерасчёта tRC для LPDDR4. 0 - перерасчет с использованием +1; 1 - перерасчет с использованием +2; 2 - перерасчет с использованием +3; 3 - перерасчет с использованием +4. Присутсвует только в реализациях, сконфигурированных для поддержки LPDDR4. Требуемые циклы для перерасчёта могут быть определены за счёт деления 3,75 нс на период core_ddrc_core_clk, и округления его вверх до следующего целого. Таким образом, в это поле регистра должно быть установлено значение выше - 1. Условия существаования: MEMC_LPDDR4==1. Режим программирования: квазидинамические группы 2 и 4 (квазидинамические группы 2 и 4).	0x0	R/W
[31:11]	–	Зарезервировано.	0	

24.6.8 Регистр интервала перерасчёта температуры (DERATEINT)

Поля регистра DERATEINT описаны в Таблица 24.10.

Таблица 24.10. Регистр интервала перерасчёта температуры

Диапазон	Название	Описание	Исходное значение	Тип доступа
[31:0]	mr4_read_interval	Интервал между двумя чтениями MR4, используемый для перерасчёта параметров задержки. Присутствует только в реализациях, сконфигурированных для поддержки LPDDR2/LPDDR3/LPDDR4. В этот регистр не должно быть установлено значение «0». Единицы измерения: такты синхросигнада DFI. Режим программирования: статический.	0x800000	R/W

24.6.9 Регистр управления перерасчётом температуры (DERATECTL)

Поля регистра DERATECTL описаны в Таблица 24.11.

Таблица 24.11. Регистр управления перерасчётом температуры

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	derate_temp_limit_intr_en	Бит подключения прерывания для контакта вывода derate_temp_limit_intr. 1 - подключен; 0 - отключен. Режим программирования: динамический.	0x1	R/W
[1]	derate_temp_limit_intr_clr	Бит очистки прерывания для derate_temp_limit_intr. В конце операции очистки прерывания, uMCTL2 автоматически обнуляет этот бит.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Контролепригодность: readOnly. Режим программирования: динамические.		
[2]	derate_temp_limit_intr_force	Бит принудительного прерывания (Interrupt force bit) для derate_temp_limit_intr. Установка 1 в это поле спровоцирует установку сигнала на контакте вывода derate_temp_limit_intr. В конце операции принудительного прерывания uMCTL2 автоматически обнуляет этот бит. Контролепригодность: readOnly. Режим программирования: динамический.	0x0	R/W
[31:3]	–	Зарезервировано.	0	

24.6.10 Регистр управления низким питанием (PWRCTL)

Поля регистра PWRCTL описаны в Таблица 24.12.

Таблица 24.12. Регистр управления низким питанием

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	selfref_en	Если это поле истинно, то uMCTL2 помещает SDRAM в состояние саморегенерации после программируемого количества циклов «максимального числа тактов в ждущем режиме перед саморегенерацией (PWRTMG.selfref_to_x32)». Этот бит регистра можно перепрограммировать в процессе стандартной работы. Режим программирования: динамический.	0x0	R/W
[1]	powerdown_en	Если это поле истинно, uMCTL2 уходит в режим низкого питания после программируемого количества циклов «максимального числа тактов в ждущем режиме перед саморегенерацией (PWRTMG.powerdown_to_x32)». Этот бит регистра можно перепрограммировать в процессе стандартной работы. Режим программирования: динамический.	0x0	R/W
[2]	deeppowerdown_en	Когда это поле имеет значение 1, uMCTL2 помещает SDRAM в глубокий режим низкого питания, когда в хранилище транзакций пусто. Этот регистр должен быть сброшен до «0», чтобы вывести uMCTL2 из глубокого режима низкого питания. Контроллер производит автоматическую инициализацию SDRAM на выходе из глубокого режима низкого питания. Присутствует только в реализациях, сконфигурированных для поддержки mDDR или LPDDR2 или LPDDR3. Для реализаций, не относящихся к mDDR/LPDDR2/LPDDR3, в это поле не следует выставлять 1. ИСПОЛЬЗОВАТЬ ТОЛЬКО В РАБОЧЕМ РЕЖИМЕ. Режим программирования: динамический.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
[3]	en_dfi_dram_clk_disable	Подключает установку dfi_dram_clk_disable когда синхросигнал не требуется для SDRAM. Если установлен «0», dfi_dram_clk_disable не устанавливается никогда. Установка dfi_dram_clk_disable производится при следующих условиях: - в DDR2/DDR3 может быть установлено только в режиме саморегенерации; - в DDR4 может быть установлен в следующих состояниях: 1) в режиме саморегенерации; 2) в режиме максимального энергосбережения (Maximum Power Saving Mode); - в mDDR/LPDDR2/LPDDR3 может быть установлен в следующих состояниях: 1) в режиме саморегенерации; 2) в режиме низкого питания (Power Down); 3) в глубоком режиме низкого питания (Deep Power Down); 4) в стандартном режиме работы (остановка синхросигнала, Clock Stop); - в LPDDR4 может быть установлен в следующих состояниях: 1) во время саморегенерации в режиме низкого питания (Self Refresh Power Down); 2) в режиме низкого питания (Power Down); 3) в стандартном режиме работы (остановка синхросигнала, Clock Stop); Режим программирования: динамический.	0x0	R/W
[4]	mpsm_en	Когда это поле имеет значение «1», uMCTL2 помещает SDRAM в режим максимального энергосбережения, когда в хранилище транзакций пусто. Этот регистр должен быть сброшен до «0», чтобы вывести uMCTL2 из режима максимального энергосбережения. Присутствует только в реализациях, сконфигурированных для поддержки DDR4. Для реализаций, не затрагивающих DDR4, в это поле регистра не следует устанавливать 1. Следует отметить, что MPSM не поддерживается, когда используется DDR PHY от Synopsys DWC, если параметр PHY DWC_AC_CS_USE отключен, поскольку алгоритм выхода из MPSM требует, чтобы сигнал "выбор микросхемы" (chip-select signal) переключался (toggle). ИСПОЛЬЗОВАТЬ ТОЛЬКО В РАБОЧЕМ РЕЖИМЕ. Режим программирования: динасический.	0x0	R/W
[5]	selfref_sw	Установка «1» в это поле провоцирует систему немедленно переключиться в режим саморегенерации, если только она не находится в состоянии инициализации (INIT) или в рабочем режиме DPD/MPSM.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Этот бит известен как программный вход/выход для состояния саморегенерации (Software Entry/Exit to Self-Refresh) 1 - программный вход в состояние саморегенерации; 2 - программный выход из состояния саморегенерации; Режим программирования: динамический.		
[6]	stay_in_selfref	Состояние саморегенерации - промежуточное состояние перед входом в состояние саморегенерации при низком питании (Self-refresh power down) или выходом из состояния саморегенерации при низком питании для LPDDR4. Это поле управляет переключением из состояния саморегенерации: 1 - запрещает переключение из состояния саморегенерации; 0 - разрешает переключение из состояния саморегенерации; Режим программирования: динамический.	0x0	R/W
[7]	dis_cam_drain_selfref	Указывает, разрешено ли пропускать очищение памяти CAM (Content Addressable Memory - ассоциативной памяти) при входе в состояние саморегенерации. Это поле не может быть модифицировано до тех пор, пока PWRCTL.selfref_sw==1. 0 - блоки памяти CAMs должны быть пусты перед входом в состояние саморегенерации (SR); 1 - блоки памяти CAMs не опустошаются перед входом в состояние саморегенерации (SR) (не поддерживается). Следует отметить, что PWRCTL.dis_cam_drain_selfref=1 не поддерживается в этой версии. Требуется установка PWRCTL.dis_cam_drain_selfref=0. Режим программирования: динамический.	0x0	R/W
[8]	lpddr4_sr_allowed	Указывает, разрешен ли переход из состояния саморегенерации в режиме низкого питания (SR-PD) в стандартное состояние саморегенерации (SR) и обратно к состоянию саморегенерации в режиме низкого питания (SR-PD). Это поле не может быть изменено до тех пор, пока PWRCTL.selfref_sw==1. 0 - переход SR-PD -> SR -> SR-PD не разрешен; 1 - переход SR-PD -> SR -> SR-PD разрешен. Режим программирования: динамический.	0x0	R/W
[31:9]	–	Зарезервировано.	0	

24.6.11 Регистр задержек низкого питания (PWRTMG)

Поля регистра PWRTMG описаны в Таблица 24.13.

Таблица 24.13. Регистр задержек низкого питания

Диапазон	Название	Описание	Исходное значение	Тип доступа
[4:0]	powerdown_to_x32	Задает число тактов, в течение которого канал команд DDRC должен быть в неактивном состоянии, чтобы uMCTL2 автоматически поместил SDRAM в режим низкого питания. Канал команд DDRC считается неактивным, когда нет никаких незавершенных HIF-команд (команд интерфейса хоста, Host Interface). Это должно быть подключено в PWRCTL.powerdown_en. ИСПОЛЬЗОВАТЬ ТОЛЬКО В РАБОЧЕМ РЕЖИМЕ. Единицы измерения: числа, кратные 32 тактам синхросигнала DFI. Режим программирования: квазидинамическая группа 4.	0x10	R/W
[7:5]	–	Зарезервировано.	0	
[15:8]	t_dpd_x4096	Минимальная длительность режима низкого питания. Для mDDR значение в спецификации JEDEC равно 0, поскольку mDDR выходит из режима низкого питания сразу после снятия PWRCTL.deeppowerdown_en. Для LPDDR2/LPDDR3 значение из спецификации JEDEC - 500 мкс. Присутствует только в реализациях, сконфигурированных для поддержки mDDR, LPDDR2 или LPDDR3. ИСПОЛЬЗОВАТЬ ТОЛЬКО В РАБОЧЕМ РЕЖИМЕ. Единицы измерения: числа, кратные 4096 тактам синхросигнала DFI. Режим программирования: квазидинамическая группа 4.	0x20	R/W
[23:16]	selfref_to_x32	Задает число тактов, в течение которого канал команд DDRC должен быть в неактивном состоянии, чтобы uMCTL2 автоматически поместила SDRAM в режим саморегенерации. Канал команд DDRC считается неактивным, когда нет никаких незавершенных HIF-команд. Это должно быть подключено в PWRCTL.selfref_en. ИСПОЛЬЗОВАТЬ ТОЛЬКО В РАБОЧЕМ РЕЖИМЕ. Единицы измерения: числа, кратные 32 тактам синхросигнала DFI. Режим программирования: квазидинамическая группа 4.	0x40	R/W
[31:24]	–	Зарезервировано.	0	

24.6.12 Регистр аппаратного управления низким питанием (HWLPCTL)

Поля регистра HWLPCTL описаны в Таблица 24.14.

Таблица 24.14. Регистр аппаратного управления низким питанием

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	hw_lp_en	Подключение аппаратного интерфейса низкого питания (Hardware Low Power Interface). Режим программирования: квазидинамическая группа 3.	0x1	R/W
[1]	hw_lp_exit_idle_en	Когда в этот бит запрограммированна «1», контакт DDRC cactive_in_ddrc может быть использован для выхода из режима автоматической остановки синхросигнала (clock stop), автоматического понижения питания или режимов автоматической саморегенерации. Следует отметить, что он не спровоцирует выход из саморегенерации, запущенной через аппаратный интерфейс низкого питания (HLPI) и/или с помощью программного обеспечения (PWRCTL.selfref_sw). Режим программирования: статический.	0x1	R/W
[15:2]	–	Зарезервировано.	0	
[27:16]	hw_lp_idle_x32	Период аппаратного простоя. На вывод cactive_ddrc подается низкое значение, если канал команд DDRC простаивает в течение hw_lp_idle * 32 тактов и не находится при этом в режимах INIT или DPD/MPSM. Канал команд DDRC считается простаивающим, когда нет никаких незавершенных HIF-команд. Аппаратная функция простоя отключается, когда hw_lp_idle_x32=0. hw_lp_idle_x32=1 - недопустимое значение. ИСПОЛЬЗОВАТЬ ТОЛЬКО В РАБОЧЕМ РЕЖИМЕ. Единицы измерения: числа, кратные 32 тактам синхросигнала DFI. Режим программирования: статический.	0x0	R/W
[31:28]	–	Зарезервировано.	0	

24.6.13 Регистр управления регенерацией 0 (RFSHCTL0)

Поля регистра RFSHCTL0 описаны в Таблица 24.15.

Таблица 24.15. Регистр управления регенерацией 0

Диапазон	Название	Описание	Исходное значение	Тип доступа
[1:0]	–	Зарезервировано.	0	
[2]	per_bank_refresh	1 - регенерация по одному банку; 0 - регенерация всех банков. Регенерация по одному банку позволяет передачу потока данных (traffic) к остальным банкам.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Регенерация по одному банку не поддерживается всеми устройствами LPDDR2, но должна поддерживаться всеми устройствами LPDDR3/LPDDR4. Присутствует только в реализациях, сконфигурированных для поддержки LPDDR2/LPDDR3/LPDDR4. Режим программирования: статический.		
[3]	–	Зарезервировано.	0	
[9:4]	refresh_burst	Запрограммированное значение + 1 - это разрешенное количество таймаутов регенерации, которое может накопиться прежде, чем поток данных будет заблокирован и регенерации будут запущены принудительно. Закрытие страниц для выполнения регенерации - однократная потеря, которая должна быть осуществлена для каждой группы регенераций. Таким образом, исполнение регенераций в импульсе снижает потери по каждой регенерации при таком закрытии страниц. Большие значения RFSHCTL.refresh_burst немного увеличивают степень утилизации; малые значения снижают запаздывание, связанное с регенерациями, в худших случаях. 0 - единичная регенерация; 1 - 2-импульсная регенерация (burst-of-2 refresh); 7 - 8-импульсная регенерация. Подробнее о функции импульсной регенерации следует обратиться к Разделу 3.9 спецификации DDR2 JEDEC - JESD79-2F.pdf. Для DDR2/3, регенерация всегда осуществляется по рангу, а не по банку. Ранговая регенерация может быть запущена спустя 8*tREFI тактов с использованием функции импульсной регенерации. В режиме DDR4, в соответствии с функцией мелкого дробленя (Fine Granularity), в режиме 1X могут быть отложены 8 регенераций, в режиме 2X - 16 и в режиме 4X - 32. Если используются PHY-инициированные обновления, следует устанавливать RFSHCTL0.refresh_burst с осторожностью, чтобы гарантировать, что условие tRFCmax не нарушено из-за того, что PHY-инициированное обновление происходит незадолго до момента импульсной регенерации. В такой ситуации, импульс регенерации будет задержан до тех пор, пока PHY-инициированное обновление не завершится. В режиме регенерации по банкам для LPDDR2/LPDDR3/LPDDR4 (RFSHCTL0.per_bank_refresh = 1), могут быть отложены 64 регенерации. Режим программирования: динамический - зависимый от регенерации.	0x0	R/W
[11:10]	–	Зарезервировано.	0	

Диапазон	Название	Описание	Исходное значение	Тип доступа
[16:12]	refresh_to_x1_x32	Если таймер регенерации (tRFCnom, также известный как tREFI) истек хотя бы один раз, но еще не превысил RFSHCTL0.refresh_burst+1 раз, тогда может быть осуществлена упреждающая регенерация (speculative refresh). Упреждающая регенерация - это регенерация, произведенная в тот момент, когда регенерация могла бы быть полезна, но до того, как она стала совершенно необходима. Когда шина SDRAM простаивает в течение времени, заданного этим полем RFSHCTL0.refresh_to_x1_x32 и таймер регенераций истек хотя бы один раз с предыдущей регенерации, тогда производится упреждающая регенерация. Упреждающие регенерации продолжаются последовательно до тех пор, пока никаких регенераций больше нет в стеке ожидания (no refreshes pending) или до тех пор, пока новые чтения или записи отправляются к uMCTL2. ИСПОЛЬЗОВАТЬ ТОЛЬКО В РАБОЧЕМ РЕЖИМЕ. Единицы измерения: такты синхросигнала DFI или числа, кратные 32 тактам синхросигнала DFI, в зависимости от RFSHTMG.t_rfc_nom_x1_sel. Режим программирования: динамический - зависимый от регенерации (Refresh Related).	0x10	R/W
[19:17]	–	Зарезервировано.	0	
[23:20]	refresh_margin	Граничное число тактов синхросигнала DFI до истекания срока критической регенерации или страничного таймера. Критическая регенерация должна быть запущена перед достижением этого граничного числа. Рекомендуется не изменять значение, заданное здесь по умолчанию. Текущее указано как 0x2. Оно должно быть всегда меньше, чем используемое внутренним образом t_rfc_nom/32. Следует отметить, что используемое внутренним образом t_rfc_nom равно RFSHTMG.t_rfc_nom_x1_x32 * 32 в том случае, если RFSHTMG.t_rfc_nom_x1_sel=0. Если RFSHTMG.t_rfc_nom_x1_sel=1 (только для по-банковой регенерации LPDDR2/LPDDR3/LPDDR4), то используемое внутренним образом t_rfc_nom равно RFSHTMG.t_rfc_nom_x1_x32. Надо заметить, что в LPDDR2/LPDDR3/LPDDR4, используемое внутренним образом t_rfc_nom может быть разделено на 4, если подключен перерасчёт (derating) (DERATEEN.derate_enable=1). Единицы измерения: числа, кратные 32 тактам синхросигнала DFI. Режим программирования: динамический - зависимый от регенерации.	0x2	R/W
[31:24]	–	Зарезервировано.	0	

24.6.14 Регистр управления регенерацией 1 (RFSHCTL1)

Поля регистра RFSHCTL1 описаны в Таблица 24.16.

Таблица 24.16. Регистр управления регенерацией 1

Диапазон	Название	Описание	Исходное значение	Тип доступа
[11:0]	refresh_timer0_start_value_x32	Запуск таймера регенерации для ранга 0 (присутствует только в мультиранговых конфигурациях). Это полезно для разнесения регенераций по множеству рангов, чтобы помочь процессу передачи данных. Подробнее это разъясняется в Разделе «Настройки регенерации» (Refresh Controls) главы справочника по контроллеру DDR uMCTL2 DWC, посвященной архитектуре. ИСПОЛЬЗОВАТЬ ТОЛЬКО В РАБОЧЕМ РЕЖИМЕ. Единицы измерения: числа, кратные 32 тактам синхросигнала DFI. Режим программирования: динамический - зависимый от регенерации.	0x0	R/W
[15:12]	–	Зарезервировано.	0	
[27:16]	refresh_timer1_start_value_x32	Запуск таймера регенерации для ранга 1 (присутствует только в мультиранговых конфигурациях). Это полезно для разнесения регенераций по множеству рангов, чтобы помочь процессу передачи данных. Подробнее это разъясняется в Разделе «Настройки регенерации» (Refresh Controls) главы справочника по контроллеру DDR uMCTL2 DWC, посвященной архитектуре. ИСПОЛЬЗОВАТЬ ТОЛЬКО В РАБОЧЕМ РЕЖИМЕ. Единицы измерения: числа, кратные 32 тактам синхросигнала DFI. Режим программирования: динамический - зависимый от регенерации.	0x0	R/W
[31:28]	–	Зарезервировано.	0	

24.6.15 Регистр управления регенерацией 3 (RFSHCTL3)

Поля регистра RFSHCTL3 описаны в Таблица 24.17.

Таблица 24.17. Регистр управления регенерацией 3

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	dis_auto_refresh	Когда установлено '1', отключает авторегенерацию, генерируемую uMCTL2. Когда авторегенерация отключена, система SoC должна генерировать регенерации, пользуясь регистрами DBGCMD.rankn_refresh. Когда dis_auto_refresh переключается с 0 на 1, uMCTL2 немедленно заносит в план любые регенерации, находящиеся в процессе ожидания. Если подключен повтор DDR4 CRC или контроля четности (parity) (CRCPARCTL1.crc_parity_retry_enable	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		= 1), то отключение авторегенерации не поддерживается, и этот бит должен быть обнулен. Только для DDR4: если подключен режим FGR (RFSHCTL3.refresh_mode > 0), отключение авторегенерации не поддерживается, и этот бит должен быть обнулен. Это поле регистра может быть изменено в процессе работы. Режим программирования: динамический - зависимый от регенерации.		
[1]	refresh_update_level	Переключает этот сигнал (либо с 0 на 1, либо с 1 на 0), чтобы указать, что регистры регенерации обновились. refresh_update_level не должен переключаться, когда DDRC находится в состоянии сброса (core_ddrc_rstn = 0). Регистры регенерации обновляются автоматически при выходе из состояния сброса. Режим программирования: динамический.	0x0	R/W
[3:2]	–	Зарезервировано.	0	
[6:4]	refresh_mode	Мелкомодульный режим регенерации (Fine Granularity Refresh Mode): 000 - фиксированный 1x (стандартный режим); 001 - фиксированный 2x; 010 - фиксированный 4x; 101 - подключается в процессе работы 2x (не поддерживается); 110 - подключается в процессе работы 4x (не поддерживается). Все остальные значения - зарезервированы. Замечание: если RFSHCTL3.dis_auto_refresh = 1, поддерживается только режим фиксированный 1x. Замечание: режимы, подключающиеся в процессе работы, не поддерживаются в этой версии uMCTL2. Замечание: этот бит должен быть установлен в то время, когда контроллер находится в состоянии сброса или в режиме саморегенерации. Изменение этого бита во время стандартной работы недопустимо. В будущей версии uMCTL2 будет поддерживаться реализация этого поля как динамического. Замечание: это поле даёт эффект только если используется устройство DDR4 SDRAM (MSTR.ddr4 = 1). Режим программирования: квазидинамическая группа 2.	0x0	R/W
[31:7]	–	Зарезервировано.	0	

24.6.16 Регистр задержек регенерации (RFSHTMG)

Поля регистра RFSHTMG описаны в Таблица 24.18.

Таблица 24.18. Регистр задержек регенерации

Диапазон	Название	Описание	Исходное значение	Тип доступа
[9:0]	t_rfc_min	tRFC (min): минимальное время от регенерации до регенерации или активации. Когда контроллер работает в режиме 1:1, в t_rfc_min должно быть установлено RoundUp(tRFCmin/tCK). Когда контроллер работает в режиме 1:2, в t_rfc_min должно быть установлено RoundUp(RoundUp(tRFCmin/tCK)/2). В режимах LPDDR2/LPDDR3/LPDDR4: - если применяются регенерации по всем банкам, минимальное значение tRFCmin в формулах выше равняется tRFCab. - если применяются регенерации по одному банку, минимальное значение tRFCmin в формулах выше равняется tRFCpb. В режиме DDR4 значение tRFCmin в формулах выше различно, в зависмости от режима регенерации (fixed 1X,2X,4X) и плотности упаковки устройства (device density). Пользователю следует запрограммировать подходящее число, исходя из используемого значения 'refresh_mode' и используемой плотности упаковки устройства. Единицы измерения: такты синхросигнала DFI. Режим программирования: динамический - зависимый от регенерации.	0x8c	R/W
[14:10]	–	Зарезервировано.	0	
[15]	lpddr3_trefbw_en	Используется только когда подключен тип памяти LPDDR3. Следует изменять его только когда uMCTL2 находится в состоянии сброса. Определяет, следует ли использовать параметр tREFBW (он требуется для некоторых устройств LPDDR3, совместимых с более ранними версиями спецификации JEDEC по LPDDR3): 0 - параметр tREFBW не используется; 1 - параметр tREFBW используется. Режим программирования: статический.	0x0	R/W
[27:16]	t_rfc_nom_x1_x32	Средний временной интервал между регенерациями по рангам (Average time interval between refreshes per rank). (Спецификация: 7.8мкс для DDR2, DDR3 и DDR4. По mDDR, LPDDR2, LPDDR3 и LPDDR4 следует обращаться к спецификации JEDEC). Когда контроллер работает в режиме 1:1 в этот регистр следует устанавливать RoundDown(tREFI/tCK). Когда контроллер работает в режиме 1:2 в этот регистр следует устанавливать RoundDown(RoundDown(tREFI/tCK)/2). В обоих случаях выше, если RFSHTMG.t_rfc_nom_x1_sel = 0, следует делить результат выше на 32 и округлять вниз.	0x62	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Для LPDDR2/LPDDR3/LPDDR4: - если используются регенерации по всем банкам (RFSHCTL0.per_bank_refresh= 0), в расчетах выше следует использовать tREFIab. - если используются регенерации по одному банку (RFSHCTL0.per_bank_refresh = 1), в расчетах выше следует использовать tREFIpb. Для режима DDR4, значение tREFI различно, в зависимости от режима регенерации. Пользователю следует запрограммировать подходящее значение, основываясь на значении, запрограммированном в регистре режима регенерации (refresh mode register). Следует отметить, что: - RFSHTMG.t_rfc_nom_x1_x32 должно быть больше чем 0x1; - если RFSHTMG.t_rfc_nom_x1_sel == 1, RFSHTMG.t_rfc_nom_x1_x32 должно быть больше, чем RFSHTMG.t_rfc_min; - если RFSHTMG.t_rfc_nom_x1_sel == 0, RFSHTMG.t_rfc_nom_x1_x32 * 32 должно быть больше, чем RFSHTMG.t_rfc_min; - в режимах, не относящихся к DDR4 или в режиме DDR4 “Fixed 1x”: RFSHTMG.t_rfc_nom_x1_x32 должно быть меньше или равным 0xFFE. - в режиме DDR4 “Fixed 2x”: RFSHTMG.t_rfc_nom_x1_x32 должно быть меньше или равным 0x7FF. - в режиме DDR4 “Fixed 4x”: RFSHTMG.t_rfc_nom_x1_x32 должно быть меньше или равным 0x3FF. Единицы измерения: такты синхросигнала DFI или числа, кратные 32 тактам синхросигнала DFI, в зависимости от RFSHTMG.t_rfc_nom_x1_sel. Режим программирования: динамический - зависимый от регенерации.		
[30:28]	–	Зарезервировано.	0	
[31]	t_rfc_nom_x1_sel	Определяет то, какие значения заданы в полях RFSHTMG.t_rfc_nom_x1_x32 и RFSHCTL0.refresh_to_x1_x32: x1 или x32. Режим программирования: динамический - зависимый от регенерации.	0x0	R/W

24.6.17 Регистр задержек регенерации 1 (RFSHTMG1)

Поля регистра RFSHTMG1 описаны в Таблица 24.19.

Таблица 24.19. Регистр задержек регенерации 1

Диапазон	Название	Описание	Исходное значение	Тип доступа
[15:0]	–	Зарезервировано.	0	

Диапазон	Название	Описание	Исходное значение	Тип доступа
[23:16]	t_pbr2pbr	LPDDR4: tpbR2pbR. Время между переходом от по-банковой регенерации одного банка к по-банковой регенерации другого банка (Per-bank Refresh to Per-bank refresh different bank Time). Когда контроллер работает в режиме соотношения частот 1:1, следует программировать в это поле RoundUp(tpbR2pbR/tCK). Когда контроллер работает в режиме соотношения частот 1:2, следует программировать в это поле RoundUp(RoundUp(tpbR2pbR/tCK)/2). Значение tpbR2pbR в равенствах выше различно в зависимости от плотности упаковки устройства (device density). Пользователю следует запрограммировать подходящее число из спецификации. Регистр валиден только в режиме по-банковой регенерации LPDDR4 (RFSHCTL0.per_bank_refresh_mode == 1). Режим программирования: динамический - зависимый от регенерации.	0x8c	R/W
[31:24]	–	Зарезервировано.	0	

24.6.18 Регистр конфигурации ECC 0 (ECCCFG0)

Поля регистра ECCCFG0 описаны в Таблица 24.19.

Таблица 24.20. Регистр конфигурации ECC 0

Диапазон	Название	Описание	Исходное значение	Тип доступа
[2:0]	ecc_mode	Индикатор режима ECC. 000 – ECC отключен 100 – ECC включен – SEC/DED 101 – ECC включен – расширенный ECC Остальные конфигурации зарезервированы. Режим программирования: статический.	0x0	R/W
[3]	–	Зарезервировано.	0	R
[4]	dis_scrub	Отключение скраббера ECC. Значение валидно, только когда ECCCFG0.ecc_mode = 3’b100 или 3’b101 и MEMC_USE_RMW определен. Скраббер не поддерживается с Inline ECC – значение регистра игнорируется. Режим программирования: статический.	0x0	R/W
[5]	–	Зарезервировано.	0	R
[6]	ecc_ap_en	Включение опции защиты адреса. Поддерживается только с включенным Inline ECC. 0 – выключена 1 – включена Режим программирования: статический.	0x1	R/W
[7]	ecc_region_remap_en	Разрешение маппирования области ECC. 0 – запрещено 1 – разрешено	0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Режим программирования: статический.		
[14:8]	ecc_region_map	Настройка области защищаемой памяти. Адресное пространство делится на 8/16/32/64 областей в соответствии с ECCCFG1.ecc_region_map_granu. Замечание: старшая 1/8 адресного пространства всегда используется для ECC. Младшие 7 областей являются выбираемыми защищаемыми областями. Включение/выключение защиты области задается ECCCFG0.ecc_region_map[6:0]. Оставшиеся верхние области (если есть) являются незащищаемыми. Каждый бит ECCCFG0.ecc_region_map[6:0] соответствует 7 младшим областям. Для защиты области с помощью ECC следует установить соответствующий бит в 1. Все 0 являются некорректной комбинацией – должна быть хотя бы одна защищаемая область, если inline ECC разрешен в регистре ECCCFG0.ecc_mode. Для защиты всех областей следует выбрать: ecc_region_map=7'b1111111 ecc_region_map_granu=0 Только первая 1/64 защищается при следующей настройке: ecc_region_map=7'b0000001 ecc_region_map_granu=3 Режим программирования: квазидинамическая группа 3.	0x7f	R/W
[15]	–	Зарезервировано.	0	R
[21:16]	blk_channel_idle_time_x32	Устанавливает количество тактов интерфейса HIF без обращений, по истечении которых происходит сохранение всех в память всех контрольных бит. Для сохранения контрольных бит генерируется команда записи ECC (при отсутствии входящих команд интерфейса HIF), если имеются ожидающие записи блоки контрольных бит. 0 – нет таймаута (не поддерживается в текущей версии) 1 – 32 такта 2 – 64 такта и т.д. Единица измерения – 32 такта синхросигнала DFI. Режим программирования: квазидинамическая группа 3.	0x3f	R/W
[23:22]	–	Зарезервировано.	0	R
[26:24]	ecc_ap_err_threshold	Задание порога количества ошибок четности адреса. ECCAPSTAT.ecc_ap_err устанавливается при превышении количества ошибок (исправляемых/неисправляемых) внутри одной посылки. Значение этого регистра должно быть меньше, чем «Общее количество проверок ECC внутри одной посылки», при использовании данной опции. «Общее количество проверок ECC внутри одной посылки» вычисляется как «ширина DRAM» * «DRAM Burst Length» / 64. Режим программирования: статический	0x3	R/W
[28:27]	–	Зарезервировано.	0	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
[29]	ecc_region_map_other	Когда ECCCFG0.ecc_region_map_granu>0, существует область, не контролируемая ecc_region_map. Регистр задает режим защиты для Inline ECC. 0 – не защищен 1 – защищен Регистр валиден только при ECCCFG0.ecc_region_map_granu>0 и ECCCFG0.ecc_mode=4. Режим программирования: статический	0	R/W
[31:30]	ecc_region_map_granu	Управление шагом защищаемой области. Следует задать один из возможных вариантов разбиения: 0 - 1/8 адресного пространства 1 - 1/16 адресного пространства 2 - 1/32 адресного пространства 3 - 1/64 адресного пространства Режим программирования: статический	0x0	R/W

24.6.19 Регистр конфигурации ECC 1(ECCCFG1)

Поля регистра ECCCFG1 описаны в Таблица 24.19.

Таблица 24.21. Регистр ECCCFG1

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	data_poison_en	Разрешение внесения ошибок в данные, записываемые по адресу ECCPOISONADDRE0/1. Режим программирования: квазидинамическая группа 3.	0x0	R/W
[1]	data_poison_bit	Определяет количество вносимых ошибок. 0 – 2 ошибки (некорректируемая ошибка) 1 – 1 ошибка (корректируемая ошибка) Режим программирования: квазидинамическая группа 3.	0x0	R/W
[3:2]	–	Зарезервировано.	0	R
[4]	ecc_region_parity_lock	Блокировка дроступа к секции, хранящей биты четности ECC. 1 – блокировка: при доступе к региону генерируется отклик с ошибкой. 0 – обычный доступ к области к незащищенной ECC. Режим программирования: квазидинамическая группа 3.	0x1	R/W
[5]	ecc_region_waste_lock	Блокировка дроступа к оставшейся части секции, хранящей биты четности ECC. 1 – блокировка: при доступе к региону генерируется отклик с ошибкой. 0 – обычный доступ к области к незащищенной ECC. Режим программирования: квазидинамическая группа 3.	0x1	R/W
[6]	–	Зарезервировано.	0	R
[7]	blk_channel_active_term	Разрешение прерывания активного блочного канала. 0 – запрещено: используется только для отладки 1 – разрешено: блок освобождается, когда все адреса внутри блока записаны или прочитаны Режим программирования: статический.	0x1	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
[11:8]	active_blk_channel	Количество активных блочных каналов. Указывает число различных областей HIF адресов, которые хранятся в контроллере. Общее количество блочных каналов ECC равно 4. Регистр позволяет программно уменьшать количество активных каналов числом, равным значению регистра +1. Режим программирования: квазидинамическая группа 3.	0x3	R/W
[31:12]	–	Зарезервировано.	0	R

24.6.20 Регистр состояния ECCSTAT (ECCSTAT)

Поля регистра ECCSTAT описаны в Таблица 24.19.

Таблица 24.22. Регистр ECCSTAT

Диапазон	Название	Описание	Исходное значение	Тип доступа
[6:0]	ecc_corrected_bit_num	Позиция бита в линии при однобитной ошибке. Регистр содержит до 72 бит данных, относящихся к линии. Если ошибки в нескольких линиях, значение младшей линии отображается в данном регистре. Режим программирования: динамический.	0x0	R
[7]	–	Зарезервировано.	0	R
[8]	ecc_corrected_err	Признак наличия однобитной ошибки на любой линии. Режим программирования: динамический.	0x0	R
[15:9]		Зарезервировано.	0	R
[16]	ecc_uncorrected_er	Признак наличия двухбитной ошибки на любой линии. Режим программирования: динамический.	0x0	R
[31:17]	–	Зарезервировано.	0	R

24.6.21 Регистр очистки ECC (ECCCTL)

Поля регистра ECCCTL описаны в Таблица 24.19.

Таблица 24.23. Регистр ECCCTL

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	ecc_corrected_err_clr	Запись 1 приводит к очистке информации о корректируемых ошибках ECC. Регистр сбрасывается аппаратно. Следующие регистры очищаются: ECCSTAT.ecc_corrected_err ADVECCSTAT.advecc_corrected_err ADVECCSTAT.advecc_num_err_symbol ADVECCSTAT.advecc.err_symbol_pos ADVECCSTAT.advecc.err_symbol_bits ECCCSYN0 ECCCSYN1 ECCCSYN2 ECCBITMASK0 ECCBITMASK1 ECCBITMASK2	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Режим программирования: динамический.		
[1]	ecc_uncorrected_err_clr	Запись 1 приводит к очистке информации о некорректируемых ошибках ECC. Регистр сбрасывается аппаратно. Следующие регистры очищаются: ECCSTAT.ecc_uncorrected_err ADVECCSTAT.advecc_uncorrected_err ECCUSYN0 ECCUSYN1 ECCUSYN2 Режим программирования: динамический.	0x0	R/W
[2]	ecc_corr_err_cmt_clr	Запись 1 приводит к очистке счетчика корректируемых ошибок ECCERRCNT.ecc_corr_err_cnt. Регистр сбрасывается аппаратно. Режим программирования: динамический.	0x0	R/W
[3]	ecc_uncorr_err_cnt_clr	Запись 1 приводит к очистке счетчика некорректируемых ошибок ECCERRCNT.ecc_uncorr_err_cnt. Регистр сбрасывается аппаратно. Режим программирования: динамический.	0x0	R/W
[4]	ecc_ap_err_intr_clr	Очистка бита прерывания ecc_ap_err. Установка бита приводит к очистке ECCAPSTAT.ecc_ap_err/ecc_ap_err_intr. Бит сбрасывается аппаратно. Режим программирования: динамичкский.	0x0	R/W
[7:5]	–	Зарезервировано.	0	R
[8]	ecc_corrected_err_intr_en	Разрешение прерывания для ecc_corrected_err_intr. 1 – разрешено 0 - запрещено Режим программирования: динамический.	0x1	R/W
[9]	ecc_uncorrected_err_intr_en	Разрешение прерывания для ecc_uncorrected_err_intr. 1 – разрешено 0 - запрещено Режим программирования: динамический.	0x1	R/W
[10]	ecc_ap_err_intr_en	Разрешение прерывания для ecc_ap_err_intr. 1 – разрешено 0 - запрещено Режим программирования: динамический.	0x1	R/W
[15:11]	–	Зарезервировано.	0	R
[16]	ecc_corrected_err_intr_force	Генерация прерывания ecc_corrected_err_intr. Установка бита приводит к генерации прерывания. Бит сбрасывается аппаратно. Режим программирования: динамический.	0x0	R/W
[17]	ecc_uncorrected_err_intr_force	Генерация прерывания ecc_uncorrected_err_intr. Установка бита приводит к генерации прерывания. Бит сбрасывается аппаратно. Режим программирования: динамический.	0x0	R/W
[18]	ecc_app_err_ints_force	Генерация прерывания ecc_ap_err_intr. Установка бита приводит к генерации прерывания. Бит сбрасывается аппаратно. Режим программирования: динамический.	0x0	R/W
[31:19]	–	Зарезервировано.	0	R

24.6.22 Регистр счетчика ошибок (ECCERRCNT)

Поля регистра ECCERRCNT описаны в Таблица 24.19.

Таблица 24.24. Регистр ECCERRCNT

Диапазон	Название	Описание	Исходное значение	Тип доступа
[15:0]	ecc_corr_err_cnt	Количество обнаруженных единичных ошибок ECC. Останавливается при достижении значения 0xFFFF. Режим программирования: динамический.	0x0	R
[31:16]	ecc_uncorr_err_cnt	Количество обнаруженных неисправляемых ошибок ECC. Останавливается при достижении значения 0xFFFF. Режим программирования: динамический.	0x0	R

24.6.23 Регистр адреса корректируемой ошибки 0 (ECCCADDR0)

Поля регистра ECCCADDR0 описаны в Таблица 24.19.

Таблица 24.25. Регистр ECCCADDR0

Диапазон	Название	Описание	Исходное значение	Тип доступа
[17:0]	ecc_corr_row	Номер страницы/ряда, чтение из которого привело к единичной ошибке. Режим программирования: динамический.	0x0	R
[23:18]	–	Зарезервировано.	0	R
[24]	ecc_corr_rank	Номер ранка, чтение из которого привело к корректируемой ошибке. Режим программирования: динамический.	0x0	R
[31:25]	–	Зарезервировано.	0	R

24.6.24 Регистр адреса корректируемой ошибки 1 (ECCCADDR1)

Поля регистра ECCCADDR1 описаны в Таблица 24.19.

Таблица 24.26. Регистр ECCCADDR1

Диапазон	Название	Описание	Исходное значение	Тип доступа
[11:0]	ecc_corr_col	Номер столбца, чтение из которого привело к корректируемой ошибке. Младший бит не записывается. Режим программирования: динамический.	0x0	R
[15:12]	–	Зарезервировано.	0	R
[18:16]	ecc_corr_bank	Номер банка, чтение из которого привело к корректируемой ошибке. Режим программирования: динамический.	0x0	R
[23:19]	–	Зарезервировано.	0	R
[25:24]	ecc_corr_bg	Номер группы банков, чтение из которого привело к корректируемой ошибке. Режим программирования: динамический.	0x0	R
[31:26]	–	Зарезервировано.	0	R

24.6.25 Регистр синдрома корректируемой ошибки 0 (ECCCSYN0)

Поля регистра ECCCSYN0 описаны в Таблица 24.19.

Таблица 24.27. Регистр ECCCSYN0

Диапазон	Название	Описание	Исходное значение	Тип доступа
[31:0]	ecc_corr_syndromes_31_0	Значение данных, приведшее к корректируемой ошибке. Для 16-битного ECC используются только младшие 16 бит. Режим программирования: динамический.	0x0	R

24.6.26 Регистр синдрома корректируемой ошибки 1 (ECCCSYN1)

Поля регистра ECCCSYN1 описаны в Таблица 24.19.

Таблица 24.28. Регистр ECCCSYN1

Диапазон	Название	Описание	Исходное значение	Тип доступа
[31:0]	ecc_corr_syndromes_63_32	Значение данных, приведшее к корректируемой ошибке. Для 16-битного и 32-битного ECC регистр не используется. Режим программирования: динамический.	0x0	R

24.6.27 Регистр синдрома корректируемой ошибки 2 (ECCCSYN2)

Поля регистра ECCCSYN2 описаны в Таблица 24.19.

Таблица 24.29. Регистр ECCCSYN2

Диапазон	Название	Описание	Исходное значение	Тип доступа
[7:0]	ecc_corr_syndromes_71_64	Значение данных, приведшее к корректируемой ошибке. Регистр относится к проверочным битам. Для 64-битного ECC [71:64], для 32-битного [39:32], для 16-битного [23:16]. Режим программирования: динамический.	0x0	R
[31:8]		Заразервировано	0	R

24.6.28 Регистр маски корректируемых бит 0 (ECCBITMASK0)

Поля регистра ECCBITMASK0 описаны в Таблица 24.19.

Таблица 24.30. Регистр ECCBITMASK0

Диапазон	Название	Описание	Исходное значение	Тип доступа
[31:0]	ecc_corr_bit_mask_31_0	Маска корректируемых бит. 1 в любой позиции указывает, что для этой позиции была обнаружена исправимая ошибка. Регистр накапливает ошибки с целью общей индикации проблемных битовых линий. Режим программирования: динамический.	0x0	R

24.6.29 Регистр маски корректируемых бит 1 (ECCBITMASK1)

Поля регистра ECCBITMASK1 описаны в Таблица 24.19.

Таблица 24.31. Регистр ECCBITMASK1

Диапазон	Название	Описание	Исходное значение	Тип доступа
[31:0]	ecc_corr_bit_mask_63_32	Маска корректируемых бит. 1 в любой позиции указывает, что для этой позиции была обнаружена исправимая ошибка. Регистр накапливает ошибки с целью общей индикации проблемных битовых линий. Режим программирования: динамический.	0x0	R

24.6.30 Регистр маски корректируемых бит 2 (ECCBITMASK2)

Поля регистра ECCBITMASK2 описаны в Таблица 24.19.

Таблица 24.32. Регистр ECCBITMASK2

Диапазон	Название	Описание	Исходное значение	Тип доступа
[7:0]	ecc_corr_bit_mask_71_64	Маска корректируемых бит, относящийся к контрольным битам. 1 в любой позиции указывает, что для этой позиции была обнаружена исправимая ошибка. Регистр накапливает ошибки с целью общей индикации проблемных битовых линий. Режим программирования: динамический.	0x0	R
[31:8]	–	Зарезервировано.	0x0	R

24.6.31 Регистр адреса некорректируемой ошибки 0 (ECCUADDR0)

Поля регистра ECCUADDR0 описаны в Таблица 24.19.

Таблица 24.33. Регистр ECCUADDR0

Диапазон	Название	Описание	Исходное значение	Тип доступа
[17:0]	ecc_uncorr_row	Номер страницы/ряда, чтение из которого привело к некорректируемой ошибке. Режим программирования: динамический.	0x0	R
[23:18]	–	Зарезервировано.	0	R
[24]	ecc_uncorr_rank	Номер ранка, чтение из которого привело к некорректируемой ошибке. Режим программирования: динамический.	0x0	R
[31:25]	–	Зарезервировано.	0	R

24.6.32 Регистр адреса некорректируемой ошибки 1 (ECCUADDR1)

Поля регистра ECCUADDR1 описаны в Таблица 24.19.

Таблица 24.34. Регистр ECCUADDR1

Диапазон	Название	Описание	Исходное значение	Тип доступа
[11:0]	ecc_uncorr_col	Номер столбца, чтение из которого привело к некорректируемой ошибке. Младший бит не записывается.	0x0	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Режим программирования: динамический.		
[15:12]	–	Зарезервировано.	0	R
[18:16]	ecc_uncorr_bank	Номер банка, чтение из которого привело к некорректируемой ошибке. Режим программирования: динамический.	0x0	R
[23:19]	–	Зарезервировано.	0	R
[25:24]	ecc_uncorr_bg	Номер группы банков, чтение из которого привело к некорректируемой ошибке. Режим программирования: динамический.	0x0	R
[31:26]	–	Зарезервировано.	0	R

24.6.33 Регистр синдрома некорректируемой ошибки 0 (ECCUSYN0)

Поля регистра ECCUSYN0 описаны в Таблица 24.19.

Таблица 24.35. Регистр ECCUSYN0

Диапазон	Название	Описание	Исходное значение	Тип доступа
[31:0]	ecc_uncorr_syndromes_31_0	Значение данных, приведшее к некорректируемой ошибке. Для 16-битного ECC используются только младшие 16 бит. Режим программирования: динамический.	0x0	R

24.6.34 Регистр синдрома некорректируемой ошибки 1 (ECCUSYN1)

Поля регистра ECCUSYN1 описаны в Таблица 24.19.

Таблица 24.36. Регистр ECCUSYN1

Диапазон	Название	Описание	Исходное значение	Тип доступа
[31:0]	ecc_uncorr_syndromes_63_32	Значение данных, приведшее к некорректируемой ошибке. Не используется для 16-битного и 32-битного ECC. Режим программирования: динамический.	0x0	R

24.6.35 Регистр синдрома некорректируемой ошибки 2 (ECCUSYN2)

Поля регистра ECCUSYN2 описаны в Таблица 24.19.

Таблица 24.37. Регистр ECCUSYN2

Диапазон	Название	Описание	Исходное значение	Тип доступа
[7:0]	ecc_uncorr_syndromes_71_64	Значение данных, приведшее к некорректируемой ошибке, относящееся к проверочным битам. Для 64-битного ECC [71:64], для 32-битного [39:32], для 16-битного [23:16]. Режим программирования: динамический.	0x0	R
[31:8]	–	Зарезервировано.	0	R

24.6.36 Регистр адреса внесенной ошибки 0 (ECCPOISONADDR0)

Поля регистра ECCPOISONADDR0 описаны в Таблица 24.19. Если адрес трнзакции совпадает с адресом ECCPOISONADDRx, ошибка ECC будет добавлена в транзакцию при ECCCFG1.data_poison_en = 1.

Таблица 24.38. Регистр ECCPOISONADDR0

Диапазон	Название	Описание	Исходное значение	Тип доступа
[11:0]		Адрес столбца для внесения ошибки. Адрес должен быть выровнен с транзакцией. В режиме полной ширины шины ecc_poison_col[2:0] = 0. В режиме половинной ширины шины ecc_poison_col[3:0] = 0. В режиме четвертной ширины шины ecc_poison_col[4:0] = 0. Режим программирования: статический.	0x0	
[23:12]	–	Зарезервировано.	0	R
[24]		Адрес ранка для внесения ошибки. Режим программирования: статический.	0x0	
[31:25]	–	Зарезервировано.	0	R

24.6.37 Регистр адреса внесенной ошибки 1 (ECCPOISONADDR1)

Поля регистра ECCPOISONADDR1 описаны в Таблица 24.19.

Таблица 24.39. Регистр ECCPOISONADDR1

Диапазон	Название	Описание	Исходное значение	Тип доступа
[17:0]		Адрес ряда/страницы для внесения ошибки. Режим программирования: статический.	0x0	R/W
[23:18]	–	Зарезервировано.	0	R
[26:24]		Адрес банка для внесения ошибки. Режим программирования: статический.	0x0	R/W
[27]	–	Зарезервировано.	0	R
[29:28]		Адрес группы банков для внесения ошибки. Режим программирования: статический.	0x0	R/W
[31:30]	–	Зарезервировано.	0	R

24.6.38 Регистр управления контролем четности и CRC 0 (CRCPARCTL0)

Поля регистра CRCPARCTL0 описаны в Таблица 24.40.

Таблица 24.40. Регистр управления контролем четности и CRC 0

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	dfi_alert_err_int_en	Бит подключения прерывания для аварийной ошибки DFI (DFI alert error). Если этот бит установлен, любая ошибка контроля четности/CRC, обнаруженная на	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		вводе dfi_alert_n приведет к установке прерывания на CRCPARSTAT.dfi_alert_err_int. Режим программирования: динамический.		
[1]	dfi_alert_err_int_clr	Бит очистки прерывания для аварийной ошибки DFI (DFI alert error). Если этот бит установлен, прерывание аварийной ошибки на CRCPARSTAT.dfi_alert_err_int будет очищено. uMCTL2 очищает этот бит автоматически. Контролепригодность: readOnly. Режим программирования: динамический.	0x0	R/W
[2]	dfi_alert_err_cnt_clr	Очистка счетчика аварийной ошибки DFI. Бит очистки счетчика аварийной ошибки DFI. Установка этого бита очистит счтечик аварийной ошибки DFI, CRCPARSTAT.dfi_alert_err_cnt. uMCTL2 очищает этот бит автоматически. Контролепригодность: readOnly. Режим программирования: динамический.	0x0	R/W
[31:3]	–	Зарезервировано.	0	

24.6.39 Регистр управления контролем четности и CRC 1 (CRCPARCTL1)

Поля регистра CRCPARCTL1 описаны в Таблица 24.41.

Таблица 24.41. Регистр управления контролем четности и CRC 1

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	parity_enable	Поле подключения контроля четности команды/адреса (C/A Parity). 1: подключает генерацию контроля четности C/A и обнаружение ошибки контроля четности C/A. 0: отключает генерацию контроля четности C/A и обнаружение ошибки контроля четности C/A. Если подключено обнаружение ошибки контроля четности через RCD (Registering Clock Driver, регистрирующий формирователь синхросигнала) или обнаружение ошибки контроля четности в SDRAM, в это поле должна быть установлена 1. Режим программирования: статический.	0x0	R/W
[3:1]	–	Зарезервировано.	0	
[4]	crc_enable	Регистр подключения CRC (Cyclic redundancy check, циклический избыточный код). 1: подключает генерацию CRC; 0: отключает генерацию CRC. Установка этого поля регистра должна соответствовать установке регистра режима CRC в DRAM. Режим программирования: квазидинамическая группа 2.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
[6:5]	–	Зарезервировано.	0	
[7]	crc_inc_dm	Регистр настроек расчёта CRC (CRC Calculation setting register). 1: CRC включает и сигнал DM; 0: CRC не включает в себя сигнал DM. Присутствует только в реализациях, сконфигурированных для поддержки DDR4. Режим программирования: статический.	0x0	R/W
[11:8]	–	Зарезервировано.	0x0	
[12]	caparity_disable_before_sr	Если контроль четности команды/адреса (CA parity) для DDR4-SDRAM подключен через INIT6.mr5[2:0]!=0 и в это поле установлена «1», контроль четности CA отключается автоматически перед входом в режим саморегенерации и подключается обратно после выхода из саморегенерации путем отправки MR5. 1: контроль четности CA отключен перед входом в режим саморегенерации; 0: контроль четности CA не отключен перед входом в режим саморегенерации. Если применяется режим замедления (Geardown) через установку MSTR.geardown_mode=1, то в это поле должна быть установлена «1». Если в этот регистр установлен 0, значения DRAMTMG5.t_ckesr и DRAMTMG5.t_cksre должны быть увеличены на величину запаздывания контроля четности PL (Parity latency). Режим программирования: статический.	0x1	R/W
[31:13]	–	Зарезервировано.	0	

24.6.40 Регистр статуса контроля четности и CRC (CRCPARSTAT)

Поля регистра CRCPARSTAT описаны в Таблица 24.42.

Таблица 24.42. Регистр статуса контроля четности и CRC

Диапазон	Название	Описание	Исходное значение	Тип доступа
[15:0]	dfi_alert_err_cnt	Счетчик аварийной ошибки DFI (DFI alert error count). Если на dfi_alert_n обнаружена ошибка контроля четности/CRC, этот счетчик увеличивается на 1. Это поле не зависит от установки CRCPARCTL0.dfi_alert_err_int_en. Оно достигает заполнения на значении 0xFFFF, и может быть очищено путем установки CRCPARCTL0.dfi_alert_err_cnt_clr. Режим программирования: статический.	0x0	-

Диапазон	Название	Описание	Исходное значение	Тип доступа
[16]	dfi_alert_err_int	Прерывание аварийной ошибки DFI (DFI alert error interrupt). Если на dfi_alert_n обнаружена ошибка контроля четности/CRC, и прерывание подключено через CRCPARCTL0.dfi_alert_err_int_en, будет установлен этот бит прерывания. Он будет установлен до тех пор, пока его не очистит CRCPARCTL0.dfi_alert_err_cnt_clr. Режим программирования: статический.	0x0	-
[31:17]	–	Зарезервировано.	0	-

24.6.41 Регистр инициализации SDRAM 0 (INIT0)

Поля регистра INIT0 описаны в Таблица 24.43.

Таблица 24.43. Регистр инициализации SDRAM 0

Диапазон	Название	Описание	Исходное значение	Тип доступа
[11:0]	pre_cke_x1024	Количество тактов после сброса, которое следует прождать перед установкой высокого значения CKE для запуска последовательности инициализации SDRAM. Спецификации DDR2, как правило, требуют, чтобы это поле было запрограммированно под задержку >= 200 мкс. LPDDR2/LPDDR3: tINIT1 = 100 нс (минимально); LPDDR4: tINIT3 = 2 мс (минимально). Когда контроллер работает в режиме соотношения частот 1:2, в это поле следует программировать половину значения, заданного в спецификации JEDEC, округленную вверх до следующей целой величины. Для RDIMMs в DDR3/DDR4 это значение должно включать время, необходимое для удовлетворения условию tSTAB. Единицы измерения: числа, кратные 1024 тактам синхросигнала DFI. Режим программирования: статический.	0x4e	R/W
[15:12]	–	Зарезервировано.	0	
[25:16]	post_cke_x1024	Количество тактов, которое следует прождать после установки высокого значения CKE для запуска последовательности инициализации SDRAM. DDR2, как правило, требует задержки в 400 нс, что требует программировать в это значение «2» на всех скоростях синхросигналов. LPDDR2/LPDDR3, как правило, требуют, чтобы это поле было запрограммированно под задержку в 200 мкс. LPDDR4, как правило, требует, чтобы это поле было запрограммированно под задержку в 2 мкс.	0x2	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Когда контроллер работает в режиме соотношения частот 1:2, в это поле следует программировать половину значения, заданного в спецификации JEDEC, округленную вверх до следующей целой величины. Единицы измерения: числа, кратные 1024 тактам синхросигнала DFI. Режим программирования: статический.		
[29:26]	–	Зарезервировано.	0	
[31:30]	skip_dram_init	Если подключен нижний бит поля, алгоритм инициализации SDRAM пропускается. Верхний бит поля задает, в каком состоянии контроллер запускается после снятия сброса. 00 - алгоритм инициализации SDRAM запускается после включения питания; 01 - алгоритм инициализации SDRAM пропускается после включения питания. Контроллер запускается в стандартном режиме (Normal Mode); 11 - алгоритм инициализации SDRAM пропускается после включения питания. Контроллер запускается в режиме саморегенерации (Self-refresh Mode); 10 - зарезервировано. Режим программирования: квазидинамическая группа 2.	0x0	R/W

24.6.42 Регистр инициализации SDRAM 1 (INIT1)

Поля регистра INIT1 описаны в Таблица 24.44.

Таблица 24.44. Регистр инициализации SDRAM 1

Диапазон	Название	Описание	Исходное значение	Тип доступа
[3:0]	pre_ocd_x32	Период ожидания перед установкой команды завершения функции OCD (Off-Chip Driver, внешний формирователь) на SDRAM. Специфических требований к этому полю нет; в него можно устанавливать 0. Единицы измерения: числа, кратные 32 тактам синхросигнала 32 DFI. Режим программирования: статический.	0x0	R/W
[15:4]	–	Зарезервировано.	0	
[24:16]	dram_rstn_x1024	Число тактов, в течение которого производится установка сигнала сброса SDRAM в процессе последовательности инициализации. Это поле присутствует только для реализаций, поддерживающих устройства DDR3, DDR4 или LPDDR4. Чтобы применять его с DDR PHY от Synopsys, в это поле должна быть установлена минимум «1». Когда контроллер работает в режиме соотношения частот 1:2, в это поле следует программировать половину значения, заданного в спецификации	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		JEDEC, округленную вверх до следующей целой величины. Единицы измерения: числа, кратные 1024 тактам синхросигнала DFI. Режим программирования: статический.		
[31:25]	–	Зарезервировано.	0	

24.6.43 Регистр инициализации SDRAM 2 (INIT2)

Поля регистра INIT2 описаны в Таблица 24.45.

Таблица 24.45. Регистр инициализации SDRAM 2

Диапазон	Название	Описание	Исходное значение	Тип доступа
[3:0]	min_stable_clock_x1	Время, которое следует прождать после первой установки высокого значения CKE, tINIT2. Присутствует только в реализациях, сконфигурированных для поддержки LPDDR2/LPDDR3. LPDDR2/LPDDR3, как правило, требует задержки 5 x tCK. Когда контроллер работает в режиме соотношения частот 1:2, в это поле следует программировать половину значения, заданного в спецификации JEDEC, округленную вверх до следующей целой величины. Единицы измерения: такты синхросигнала DFI. Режим программирования: статический.	0x5	R/W
[7:4]	–	Зарезервировано.	0	
[15:8]	idle_after_reset_x32	Время бездействия после команды сброса, tINIT4. Присутствует только в реализациях, сконфигурированных для поддержки LPDDR2. Когда контроллер работает в режиме соотношения частот 1:2, в это поле следует программировать половину значения, заданного в спецификации JEDEC, округленную вверх до следующей целой величины. Единицы измерения: числа, кратные 32 тактам синхросигнала DFI. Режим программирования: статический.	0xd	R/W
[31:16]	–	Зарезервировано.	0	

24.6.44 Регистр инициализации SDRAM 3 (INIT3)

Поля регистра INIT3 описаны в Таблица 24.46.

Таблица 24.46. Регистр инициализации SDRAM 3

Диапазон	Название	Описание	Исходное значение	Тип доступа
[15:0]	emr	DDR2: значение, которое следует писать в регистр EMR. Биты 9:7 предназначены для OCD и установки в этом регистре игнорируются. uMCTL2 устанавливает эти биты соответствующим образом. DDR3/DDR4: значение, которое следует писать в регистр MR1. Следует устанавливать 0 в бит 7. Если подключено тестирование режима оценки PHY (PHY-evaluation mode training), этот бит устанавливает uMCTL2 соответствующим образом в процессе уравнивания записей (write leveling). mDDR: значение, которое следует писать в регистр EMR. LPDDR2/LPDDR3/LPDDR4: значение, которое следует писать в регистр MR2. Режим программирования: квазидинамическая группа 4.	0x510	R/W
[31:16]	mr	DDR2: значение, которое следует писать в регистр MR. Бит 8 предназначен для DLL и установка в него игнорируются. uMCTL2 устанавливает этот бит соответствующим образом. DDR3/DDR4: значение, загруженное в регистр MR0. mDDR: значение, которое следует писать в регистр MR. LPDDR2/LPDDR3/LPDDR4 - значение, которое следует писать в регистр MR1. Режим программирования: квазидинамические группы 1 и 4.	0x0	R/W

24.6.45 Регистр инициализации SDRAM 4 (INIT4)

Поля регистра INIT4 описаны в Таблица 24.47.

Таблица 24.47. Регистр инициализации SDRAM 4

Диапазон	Название	Описание	Исходное значение	Тип доступа
[15:0]	emr3	DDR2: значение, которое следует писать в регистр EMR3. DDR3/DDR4: значение, которое следует писать в регистр MR3. mDDR/LPDDR2/LPDDR3: не используется. LPDDR4: значение, которое следует писать в регистр MR13. Режим программирования: квазидинамические группы 2 и 4.	0x0	R/W
[31:16]	emr2	DDR2: значение, которое следует писать в регистр EMR2. DDR3/DDR4: значение, которое следует писать в регистр MR2.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		LPDDR2/LPDDR3/LPDDR4: значение, которое следует писать в регистр MR3. mDDR: не используется. Режим программирования: квазидинамическая группа 4.		

24.6.46 Регистр инициализации SDRAM 5 (INIT5)

Поля регистра INIT5 описаны в Таблица 24.48.

Таблица 24.48. Регистр инициализации SDRAM 5

Диапазон	Название	Описание	Исходное значение	Тип доступа
[9:0]	max_auto_init_x1024	Максимальная длительность авто-инициализации, tINIT5. Присутствует только в реализациях, сконфигурированных для поддержки LPDDR2/LPDDR3. LPDDR2/LPDDR3, как правило, требует 10 мкс. Единицы измерения: числа, кратные 1024 тактам синхросигнала DFI. Режим программирования: статический.	0x4	R/W
[15:10]	–	Зарезервировано.	0	
[23:16]	dev_zqinit_x32	Внутренняя калибровка ZQ, tZQINIT. Присутствует только в реализациях, сконфигурированных для поддержки DDR3 или DDR4 или LPDDR2/LPDDR3. DDR3, как правило, требует 512 тактов синхросигнала SDRAM. DDR4 требует 1024 такта синхросигнала SDRAM. LPDDR2/LPDDR3 требует 1 мкс. Когда контроллер работает в режиме соотношения частот 1:2, в это поле следует программировать половину значения, заданного в спецификации JEDEC, округленную вверх до следующей целой величины. Единицы измерения: числа, кратные 32 тактам синхросигнала DFI. Режим программирования: статический.	0x10	R/W
[31:24]	–	Зарезервировано.	0	

24.6.47 Регистр инициализации SDRAM 6 (INIT6)

Поля регистра INIT6 описаны в Таблица 24.49.

Таблица 24.49. Регистр инициализации SDRAM 6

Диапазон	Название	Описание	Исходное значение	Тип доступа
[15:0]	mr5	DDR4: значение, которое следует загружать в регистры MR5 SDRAM.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		LPDDR4: значение, которое следует загружать в регистры MR12 SDRAM. Режим программирования: квазидинамические группы 1 и 4.		
[31:16]	mr4	DDR4: значение, которое следует загружать в регистры MR4 SDRAM. LPDDR4: значение, которое следует загружать в регистры MR11 SDRAM. Режим программирования: квазидинамические группы 2 и 4.	0x0	R/W

24.6.48 Регистр инициализации SDRAM 7 (INIT7)

Поля регистра INIT7 описаны в Таблица 24.50.

Таблица 24.50. Регистр инициализации SDRAM 7

Диапазон	Название	Описание	Исходное значение	Тип доступа
[15:0]	mr6	DDR4: значение, которое следует загружать в регистры MR6 SDRAM. LPDDR4: значение, которое следует загружать в регистры MR14 SDRAM. Режим программирования: квазидинамическая группа 4.	0x0	R/W
[31:16]	mr22	LPDDR4: значение, которое следует загружать в регистры MR22 SDRAM. Используется только в реализациях LPDDR4. Режим программирования: регистры квазидинамической группы 4.	0x0	R/W

24.6.49 Регистр управления DIMM (DIMMCTL)

Поля регистра DIMMCTL описаны в Таблица 24.51.

Таблица 24.51. Регистр управления DIMM

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	dimm_stagger_cs_en	Подключение разнесения (staggering) для мультиранговых обращений (только для мультиранговых реализаций UDIMM, RDIMM and LRDIMM). Это не поддерживается для SDRAMs mDDR, LPDDR2, LPDDR3 или LPDDR4. Замечание: даже если этот бит установлен, он не учитывает команды MR, управляемые программным обеспечением (через MRCTRL0/MRCTRL1). В этом случае программное обеспечение ответственно за отправку их по раздельным рангам соответствующим образом.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		1: (DDR4) отправка команд MRS происходит к каждому из рангов раздельно; 1: (не-DDR4) отправка всех команд происходит к четным и нечетным рангам раздельно; 0: обращения не разделяются. Режим программирования: статический.		
[1]	dimm_addr_mirr_en	Подключение дублирования адресов (Address Mirroring Enable) (для мультиранговых реализаций UDIMM и мультиранговых реализаций DDR4 RDIMM/LRDIMM). Некоторые UDIMMs и DDR4 RDIMMs/LRDIMMs реализуют дублирование адресов для нечетных рангов, что означает, что следующий адрес, адрес банка и биты группы банков меняются местами (swap):(A3, A4), (A5, A6), (A7, A8), (BA0, BA1) и также (A11, A13), (BG0, BG1) для DDR4. Установка этого бита гарантирует, что, для обращений регистров режима в процессе последовательности автоматической инициализации, эти биты тоже меняются местами внутри uMCTL2, чтобы компенсировать эту перемену мест в UDIMM/RDIMM/LRDIMM. Вдобавок к последовательности автоматической инициализации, в случае UDIMM/RDIMM/LRDIMM в DDR4, они меняются местами во время автоматического обращения MRS, для отключения/подключения отдельной функции DDR4. Замечание: это не влияет на адрес любых других обращений к памяти или обращений к регистрам режима, управляемых программным обеспечением. Это не поддерживается для SDRAMs mDDR, LPDDR2, LPDDR3 или LPDDR4. Замечание: в случае x16 DDR4 DIMMs, вывод BG1 у MRS для нечетных рангов - такой же, как и BG0, поскольку BG1 невалиден, следовательно в поле dimm_dis_bg_mirroring должна быть установлена 1. 1 - для нечетных рангов реализует дублирование адресов для команд MRS в процессе инициализации, и дублирование адресов для любых автоматических команд MRS DDR4 (для того, чтобы использовать, если в UDIMM/RDIMM/LRDIMM тоже реализовано дублирование адресов). 0 - не реализует дублирование адресов. Режим программирования: статический.	0x0	R/W
[2]	dimm_output_inv_en	Подключение инверсии вывода (Output Inversion Enable) (только для реализаций DDR4 RDIMM/LRDIMM). DDR4 RDIMM/LRDIMM реализует функцию инверсии вывода по умолчанию, что означает, что следующий адрес, адрес банка и битов группы банков в B-сторонних (B-side) DRAMs -	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		инвертированы: A3-A9, A11, A13, A17, BA0-BA1, BG0-BG1. Установка этого бита гарантирует, что для обращений регистров режима, сгенерированных uMCTL2 в процессе последовательности автоматической инициализации и подключения отдельной функции DDR4, генерируются раздельные обращения A-сторонних и B-сторонних регистров режима. Для B-сторонних обращений регистров режима, эти биты инвертированы внтури uMCTL2, чтобы скомпенсировать эту инверсию в RDIMM/LRDIMM. Рекомендуется всегда устанавливать этот бит, если используются DDR4 RDIMMs/LRDIMMs. Замечание: это не влияет на адресацию любых других обращений к памяти, или обращений регистров режима, управляемых программным обеспечением. 1: инверсия вывода реализована для B-сторонних DRAMs. 0: инверсия вывода не реализована для B-сторонних DRAMs. Режим программирования: статический.		
[3]	mrs_a17_en	Подключение бита A17 команды MRS (Enable for A17 bit of MRS command). Бит A17 адреса регистра режима определен как RFU (зарезервированный для дальнейшего использования) и в него должен быть запрограммирован 0 во время MRS. В случае, когда в качестве DDR4 RDIMM/LRDIMM подключены DRAMs, не обладающие битом A17, и подключена инверсия вывода (Output inversion), в это поле необходимо устанавливать 0, чтобы вычисление контроля четности CA не включало бит A17. Чтобы обеспечить слаженную работу с RCD, также необходимо устанавливать 1 в DA[3] в поле F0RC08 в RCD (т.е. отключать DA[3]). Замечание: это не влияет на адрес любых других обращений к памяти, или обращений регистров режима, управляемых программным обеспечением. 1 - подключен; 0 - отключен. Режим программирования: статический.	0x0	R/W
[4]	mrs_bg1_en	Подключение бита BG1 команды MRS. Бит BG1 адреса регистра режима определен как RFU (зарезервированный для дальнейшего использования) и в него должен быть запрограммирован 0 во время MRS. В случае, когда подключены DRAMs, не обладающие битом BG1, и подключены инверсия вывода и контроль четности CA, в это поле необходимо устанавливать 0, чтобы	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		вычисления контроля четности CA не включали бит BG1. Замечание: это не влияет на адресацию любых других обращений к памяти, или обращений регистров режима, управляемых программным обеспечением. Если подключено дублирование адресов, это применяется к битам BG1 четных рангов и BG0 нечетных рангов. 1 - подключен; 0 - отключен. Режим программирования: статический.		
[5]	dimm_dis_bg_mirroring	Отключение дублирования адресов для битов BG. Когда сюда установлена 1, BG0 и BG1 НЕ меняются местами, даже если подключено дублирование адресов. Это требуется для DDR4 DIMMs с устройствами x16. 1 - BG0 и BG1 НЕ меняются местами; 0 - BG0 и BG1 меняются местами если подключено дублирование адресов. Режим программирования: статический.	0x0	R/W
[6]	lrdimm_bcom_cmd_prot	Защищает ограничения задержек (tBCW/tMRC) между последовательно идущими командами BCOM, заданными в спецификации буфера данных (Data Buffer). Когда используется DDR4 LRDIMM, в этот бит должна быть установлена 1. В противном случае, в него должен быть установлен 0. Режим программирования: статический.	0x0	R/W
[11:7]	–	Зарезервировано.	0	
[12]	rcd_weak_drive	Режим слабого управления (Weak Drive mode) для установки в RCD. Это поле используется только когда uMCTL2 отключает режим CAL (Command/Address latency, запаздывание команд/адресов). Когда режим слабого управления в RCD подключен в процессе инициализации, в это поле следует устанавливать 1. Когда RCD не используется, в это поле следует устанавливать 0. 1 - подключает режим слабого управления; 0 - отключает режим слабого управления. Режим программирования: статический.	0x0	R/W
[13]	rcd_a_output_disabled	Отключает выводы RCD к A-сторонним DRAMs. Это поле используется только когда uMCTL2 отключает режим CAL. Это значение записывается в F0RC0 DA[2] перед и после отключения режима CAL. Рекомендуется устанавливать в это поле 0 во всех случаях, кроме работы с отладкой.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		1 - отключает выводы A; 0 - подключает выводы A. Режима программирования: статический.		
[14]	rcd_b_output_disabled	Отключает выводы RCD к B-сторонним DRAMs. Это поле используется только когда uMCTL2 отключает режим CAL. Это значение записывается в F0RC0 DA[3] перед и после отключения режима CAL. Рекомендуется устанавливать в это поле ~DIMMCTL.dimm_output_inv_en во всех случаях, кроме работы с отладкой. 1 - отключает выводы B; 0 - подключает выводы B. Режима программирования: статический.	0x0	R/W
[31:15]	–	Зарезервировано.	0	

24.6.50 Регистр управления рангом (RANKCTL)

Поля регистра RANKCTL описаны в Таблица 24.52.

Таблица 24.52. Регистр управления рангом

Диапазон	Название	Описание	Исходное значение	Тип доступа
[3:0]	max_rank_rd	Присутствует только для мультиранговых конфигураций. Предыстория: чтения к одному и тому же рангу могут производиться одно за другим. Чтения к разным рангам требуют добавочной паузы, определяемой регистром RANKCTL.diff_rank_rd_gap. Это сделано для того, чтобы избежать возможных конфликтов на шине данных, и чтобы дать PHY достаточно времени для переключения задержки при смене рангов. uMCTL2 осуществляет арбитраж доступов к шине данных потактово, таким образом, после того, как чтение добавлено в расписание, следуют несколько тактов синхросигнала (заданные значением регистра RANKCTL.diff_rank_rd_gap), во время которых в расписание могут быть добавлены только чтения от того же самого ранга. Это предотвращает равный доступ к шине данных для чтений от остальных рангов. Этот параметр отражает максимальное число чтений, которое может быть добавлено в расписание последовательно для одного и того же ранга. После того, как достигнуто это число, планировщик вставляет задержку, равную RANKCTL.diff_rank_rd_gap, чтобы позволить всем рангам равную возможность быть добавленными в расписание. Более высокие значения увеличивают задействованность полосы частот, более низкие - равенство (fairness).	0xf	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Эта функция может быть отключена путем установки 0 в это поле. Когда в это поле установлен 0, контроллер будет оставаться на одном и том же ранге до тех пор, пока для него доступны команды. Минимальное программируемое значение - 0 (функция отключена), а максимальное программируемое значение - 0xF. ИСПОЛЬЗОВАТЬ ТОЛЬКО В РАБОЧЕМ РЕЖИМЕ. . Режим программирования: статический.		
[7:4]	diff_rank_rd_gap	Присутствует только для мультиранговых конфигураций. Указывает число тактов, которое длится пауза в ответах данных при выполнении последовательных чтений к ранзым рангам. Это используется для переключения задержек в PHY для соответствия требованиям рангов. Это значение должно учитывать как требования PHY, так и требования ODT. Требования PHY: tphy_rdcsgap (см. справочник по PHY, в нём указано значение tphy_rdcsgap). Если начальная часть команды чтения (read preamble) установлена как 2tCK (только для DDR4), должно быть увеличено на 1. Если конечная часть команды чтения (read postamble) установлена как 1.5tCK(только для LPDDR4), должно быть увеличено на 1. Требования ODT: значение, запрограммированное в этот регистр учитывает требование задержки отключения ODT при переключении по рангам во время чтений: diff_rank_rd_gap должен иметь минимальное значение ODTCFG.rd_odt_hold - BL/2. Когда контроллер работает в режиме 1:1, следует запрограммировать в это поле большее из требований PHY или требований ODT. Когда контроллер работает в режиме 1:2, следует запрограммировать в это поле большее из значений, деленное на 2, и округлить его вверх до следующего целого числа. Следует отметить, что, если используется DDR4-LRDIMM, следует обращаться к требованиям для задержек TRDRD в спецификации JEDEC по буферу данных DDR4 (DDR4DB01). Единицы измерения: такты синхросигнала DFI. . Режим программирования: статический.	0x6	R/W
[11:8]	diff_rank_wr_gap	Присутствует только для мультиранговых конфигураций.	0x6	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Указывает число тактов, которое длится пауза в ответах данных при выполнении последовательных записей по разным рангам. Это используется для переключения задержек в PHY, чтобы они соответствовали требованиям для рангов. Это значение должно учитывать как требования PHY, так и требования ODT. Требования PHY: tphy_wrcsgap (см. справочник по PHY, в нём указано значение tphy_wrcsgap). Если подключена функция CRC, должно быть увеличено на 1. Если начальная часть команды записи (write preamble) установлена как 2tCK (только для DDR4), должно быть увеличено на 1. Если конечная часть команды записи (write postamble) установлена как 1.5tCK (только для LPDDR4), должно быть увеличено на 1. Требования ODT: значение, запрограммированное в этот регистр, учитывает требование задержки отключения ODT во время переключения по рангам в процессе записей. Для LPDDR4, с подключенным DQ ODT, diff_rank_wr_gap должно иметь минимальное значение ODTLoff - ODTLon - BL/2 + 1. Для остальных случаев, diff_rank_wr_gap должно иметь минимальное значение ODTCFG.wr_odt_hold - BL/2. Когда контроллер работает в режиме 1:1, следует запрограммировать в это поле большее из требований PHY или требований ODT. Когда контроллер работает в режиме 1:2, следует запрограммировать в это поле большее из значений, деленное на 2 и округленное вверх до следующего целого числа. Следует отметить, что, если используется DDR4-LRDIMM, следует обращаться к требованиям для задержек TWRWR в спецификации JEDEC по буферу данных DDR4 (DDR4DB01). Единицы измерения: такты синхросигнала DFI. . Если требуется значение больше, чем 0xF, следует использовать RANKCTL.diff_rank_wr_gap_msb в качестве расширения. Максимальное значение < 0x11. Режим программирования: квазидинамическая группа 2.		
[31:12]	–	Зарезервировано.	0	

24.6.51 Регистр задержек SDRAM 0 (DRAMTMG0)

Поля регистра DRAMTMG0 описаны в Таблица 24.53.

Таблица 24.53. Регистр задержек SDRAM 0

Диапазон	Название	Описание	Исходное значение	Тип доступа
[5:0]	t_ras_min	tRAS(min): минимальное время между активацией и предзарядом к одному и тому же банку. Когда контроллер работает в режиме соотношения частот 1:2, режиме 1T, следует запрограммировать в это поле tRAS(min)/2. Без округления. Когда контроллер работает в режиме соотношения частот 1:2, режиме 2T или LPDDR4, следует запрограммировать в это поле (tRAS(min)/2) и округлить вверх до следующего целого числа. Единицы измерения: такты синахросигнала DFI. Режим программирования: квазидинамические группы 2 и 4.	0xf	R/W
[7:6]	–	Зарезервировано.	0	
[14:8]	t_ras_max	tRAS(max): максимальное время между активацией и предзарядом к одному и тому же банку. Это максимальное время, в течение которого страница может удерживаться открытой. Минимальное значение этого регистра - 1. Значение 0 невалидно. Когда контроллер работает в режиме соотношения частот 1:1, следует установить в t_ras_max RoundDown(tRAS(max)/tCK/1024). Когда контроллер работает в режиме соотношения частот 1:2, следует установить в t_ras_max RoundDown((RoundDown(tRAS(max)/tCK/1024)-1)/2). Единицы измерения: числа, кратные 1024 тактам синахросигнала DFI. Режим программирования: квазидинамические группы 2 и 4.	0x1b	R/W
[15]	–	Зарезервировано.	0	
[21:16]	t_faw	tFAW: валидно только когда присутсвуют 8 или более банков (или банков и банковых групп). В 8-банковой реализации должны быть активированы самое большее 4 банка во временном окне вывода (rolling window) тактов tFAW. Когда контроллер работает в режиме соотношения частот 1:2, следует запрограммировать в это поле tFAW/2 и округлить вверх до следующего целого числа. В 4-банковой реализации, следует устанавливать в этот регистр 0x1 вне зависимости от режима соотношения частот 1:1 или 1:2. Единицы измерения: такты синахросигнала DFI. Режим программирования: квазидинамические группы 2 и 4.	0x10	R/W
[23:22]	–	Зарезервировано.	0	
[30:24]	wr2pre	Минимальное время между записью и предзарядом к одному и тому же банку. Спецификации: WL + BL/2 + tWR = примерно 8 тактов + 15 нс = 14 тактов на частоте 400МГц и менее для более низких частот, где: WL = запаздывание записи;	0xf	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		BL = длина импульса. Эта величина должна соответствовать значению, запрограммированному в бит BL регистра режима к SDRAM. BST (терминация импульса) не поддерживается в настоящее время. tWR = время восстановления для цикла записи (Write recovery time). Эта величина берется напрямую из спецификации SDRAM. Для LPDDR2/LPDDR3/LPDDR4 следует добавлять один лишний такт для этого параметра. Когда контроллер работает в режиме соотношения частот 1:2, режиме 1T, вышеописанное значение следует делить на 2. Без округления. Когда контроллер работает в режиме соотношения частот 1:2, режиме 2T или режиме LPDDR4, вышеописанное значение следует делить на 2 и округлять вверх до следующего целого значения. Следует отметить, что в зависимости от PHY, если используется LRDIMM, может быть необходимо подстроить значение этого параметра, чтобы скомпенсировать лишний такт запаздывания через LRDIMM. Единицы измерения: такты синхросигнала DFI. Режим программирования: квазидинамические группы 1, 2 и 4.		
[31]	–	Зарезервировано.	0	

24.6.52 Регистр задержек SDRAM 1 (DRAMTMG1)

Поля регистра DRAMTMG1 описаны в Таблица 24.54.

Таблица 24.54. Регистр задержек SDRAM 1

Диапазон	Название	Описание	Исходное значение	Тип доступа
[6:0]	t_rc	tRC: минимальное время между активациями к одному и тому же банку. Когда контроллер работает в режиме соотношения частот 1:2, следует запрограммировать в это поле (tRC/2) и округлить вверх до следующего целого числа. Единицы измерения: такты синхросигнала DFI. Режим программирования: квазидинамические группы 2 и 4.	0x14	R/W
[7]	–	Зарезервировано.	0	
[13:8]	rd2pre	tRTP: минимальное время от чтения до предзаряда к одному и тому же банку. DDR2: tAL + BL/2 + max(tRTP, 2) - 2; DDR3: tAL + max (tRTP, 4); DDR4: максимум следующих двух выражений: tAL + max (tRTP, 4) или RL + BL/2 - tRP (*). mDDR: BL/2	0x4	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		LPDDR2: зависит от того, является ли она LPDDR2-S2 или LPDDR2-S4: LPDDR2-S2: BL/2 + tRTP - 1. LPDDR2-S4: BL/2 + max(tRTP,2) - 2; LPDDR3: BL/2 + max(tRTP,4) - 4; LPDDR4: BL/2 + max(tRTP,8) - 8. (*) Когда и DDR4 SDRAM и ST-MRAM используются одновременно, следует использовать значение tRP из SDRAM для вычисления. Когда контроллер работает в режиме 1:2, режиме 1T, следует делить значение выше на 2. Без округления. Когда контроллер работает в режиме 1:2, режиме 2T или режиме LPDDR4, следует делить значение выше на 2 и округлять вверх до следующего целого числа. Единицы измерения: такты синхросигнала DFI. Режим программирования: квазидинамические группы 1, 2 и 4.		
[15:14]	–	Зарезервировано.	0	
[20:16]	t_xp	tXP: минимальное время от выхода из режима низкого питания (power-down exit) до любой операции. Для DDR3 необходимо, чтобы в это поле было запрограммированно tXPDLL, если в MR0[12] выбран медленный выход из режима низкого питания. Если используется контроль чётности C/A для DDR4, в это поле следует устанавливать tXP+PL. Если выбран LPDDR4 и его спецификация содержит параметр tCKELPD, в это поле следует устаналвивать наибольшее из tXP и tCKELPD. Когда контроллер работает в режиме соотношения частот 1:2, следует запрограммировать в это поле (tXP/2) и округлять вверх до следующего целого числа. Единицы измерения: такты синхросигнала DFI. Режим программирования: квазидинамические группы 2 и 4.	0x8	R/W
[31:21]	–	Зарезервировано.	0	

24.6.53 Регистр задержек SDRAM 2 (DRAMTMG2)

Поля регистра DRAMTMG2 описаны в Таблица 24.55.

Таблица 24.55. Регистр задержек SDRAM 2

Диапазон	Название	Описание	Исходное значение	Тип доступа
[5:0]	wr2rd	DDR4: CWL + PL + BL/2 + tWTR_L. LPDDR2/3/4: WL + BL/2 + tWTR + 1. Остальные: CWL + BL/2 + tWTR. В DDR4, это поле - минимальное время от команды записи до команды чтения для одной и той же	0xd	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		группы банков. В других - минимальное от команды записи до команды чтения. Это включает время обращения шины, времена восстановления и все по-банковые, по-ранговые и глобальные ограничения. Рекомендуется обратиться к соответствующему справочнику по PHY за подробностями о том, что должно быть включено сюда. Где: CWL = запаздывание записи CAS; WL = запаздывание записи; PL = запаздывание контроля четности; BL = длина импульса. Это поле должно соответствовать значению, запрограммированному в бит BL регистра режима к SDRAM. tWTR_L = внутренняя задержка от команды записи к команде чтения для одной и той же группы банков. Она извлекается напрямую из спецификации SDRAM. tWTR = внутренняя задержка от команды записи к команде чтения. Она извлекается напрямую из спецификации SDRAM. Для работы с LPDDR2/LPDDR3/LPDDR4 следует добавлять еще один лишний такт. WTR_L должна увеличиваться на 1, если используется начальная часть записи 2tCK в DDR4. Когда контроллер работает в режиме 1:2, следует делить значение, вычисленное с использованием уравнения выше на 2, и округлять его вверх до следующего целого. Единицы измерения: такты синхросигнала DFI. Режим программирования: квазидинамические группы 1, 2 и 4.		
[7:6]	–	Зарезервировано.	0	
[13:8]	rd2wr	DDR2/3/mDDR: RL + BL/2 + 2 - WL. DDR4: RL + BL/2 + 1 + WR_PREAMBLE - WL. LPDDR2/LPDDR3: RL + BL/2 + RU(tDQSCKmax/tCK) + 1 - WL. LPDDR4(DQ ODT отключена): RL + BL/2 + RU(tDQSCKmax/tCK) + WR_PREAMBLE + RD_POSTAMBLE - WL. LPDDR4(DQ ODT подключена) : RL + BL/2 + RU(tDQSCKmax/tCK) + RD_POSTAMBLE - ODTLon - RU(tODTon(min)/tCK) + 1. Минимальное время от команды чтения до команды записи. Включает время обращения шины и всех по-банковые, по-ранговые и глобальные ограничения. Рекомендуется обратиться к соответствующему справочнику по PHY за подробностями о том, что должно быть включено сюда. Где: WL = запаздывание записи;	0x6	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		BL = длина импульса. Она должна соответствовать значению, запрограммированному в бит BL регистра режима к SDRAM. RL = запаздывание чтения = запаздывание CAS; WR_PREAMBLE = 1 (1tCK начальной части записи), 2 (2tCK начальной части записи). Это уникально для DDR4 и LPDDR4. RD_POSTAMBLE = 0.5 (0.5tCK конечной части чтения), 1.5 (1.5tCK конечной части чтения). Это уникально для LPDDR4. Для LPDDR2/LPDDR3/LPDDR4, если подключено снижение параметров, (DERATEEN.derate_enable=1), должно использоваться уменьшенное tDQSCKmax. Когда контроллер работает в режиме соотношения частот 1:2, следует делить рассчитанное выше значение на 2 и округлять вверх до следующего целого числа. Следует отметить что, в зависимости от PHY, если используется LRDIMM, может быть необходимо подстроить значение этого параметра так, чтобы скомпенсировать лишний такт запаздывания через LRDIMM. Единицы измерения: такты синхросигнала DFI. Режим программирования: квазидинамические группы 1, 2 и 4.		
[15:14]	–	Зарезервировано.	0	
[21:16]	read_latency	Установка запаздывания чтения (Set to RL). Время от чтения команды до чтения данных в интерфейсе SDRAM. В это поле должно быть выставлено запаздывание чтения (RL). Следует отметить, что в зависимости от PHY, если используется RDIMM/LRDIMM, может быть необходимо подстроить значение RL, чтобы скомпенсировать лишний такт запаздывания через RDIMM/LRDIMM. Когда контроллер работает в режиме соотношения частот 1:2, следует делить значение вычисленного выражения на 2 и округлять вверх до следующего целого числа. Это поле регистра не требуется для DDR2 и DDR3 (кроме случая, когда установлен MEMC_TRAINING), поскольку запаздывания чтения и записи DFI, заданные в DFITMG0 и DFITMG1 достаточны для этих протоколов. Для всех протоколов, вдобавок к программированию этого поля регистра, необходимо запрограммировать DFITMG0 и DFITMG1 для управления запаздываниями чтения и записи. Единицы измерения: такты синхросигнала DFI. Режим программирования: квазидинамические группы 1, 2 и 4.	0x5	R/W
[23:22]	–	Зарезервировано.	0	

Диапазон	Название	Описание	Исходное значение	Тип доступа
[29:24]	write_latency	Установка запаздывания записи (Set to WL). Время от команды записи до записи данных в интерфейсе SDRAM. В это поле должно быть выставлено запаздывание записи (WL). Для mDDR в это поле в норме должна быть выставлена 1. Следует отметить, что в зависимости от PHY, если используется RDIMM/LRDIMM, может быть необходимо подстроить значение WL, чтобы скомпенсировать лишний такт запаздывания через RDIMM/LRDIMM. Когда контроллер работает в режиме соотношения частот 1:2, следует делить значение вычисленного выражения на 2 и округлять вверх до следующего целого числа. Это поле регистра не требуется для DDR2 и DDR3 (кроме случая, когда установлен MEMC_TRAINING), поскольку запаздывания чтения и записи DFI, заданные в DFITMG0 и DFITMG1 достаточны для этих протоколов. Для всех протоколов, вдобавок к программированию этого поля регистра, необходимо запрограммировать DFITMG0 и DFITMG1 для управления запаздываниями чтения и записи. Единицы измерения: такты синахросигнала DFI. Режим программирования: квазидинамические группы 1, 2 и 4.	0x3	R/W
[31:30]	–	Зарезервировано.	0	

24.6.54 Регистр задержек SDRAM 3 (DRAMTMG3)

Поля регистра DRAMTMG3 описаны в Таблица 24.56.

Таблица 24.56. Регистр задержек SDRAM 3

Диапазон	Название	Описание	Исходное значение	Тип доступа
[9:0]	t_mod	tMOD: параметр, используемый только в DDR3 и DDR4. Такты между командой загрузки режима (load mode command) и последующей командой, не относящейся к загрузке режима (non-load mode command). Если используется контроль четности C/A для DDR4, в это поле следует установить tMOD_PAR(tMOD+PL). Если подключен режим CAL (DFITMG1.dfi_t_cmd_lat > 0), время tCAL (=DFITMG1.dfi_cmd_lat) должно быть добавлено к вычислениям выше. Если используются записи MPR для DDR4, в это поле следует устанавливать tMOD + AL (или	0xc	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		tMPD_PAR + AL если также используется контроль четности C/A). Следует устанавливать в это поле tMOD если контроллер работает в режиме соотношения частот 1:1, или tMOD/2 (округленное вверх до следующего целого) если контроллер работает в режиме соотношения частот 1:2. Следует отметить что, если используется RDIMM/LRDIMM, то, в зависимости от PHY, может быть необходимо подстроить значение этого параметра так, чтобы скомпенсировать лишний такт запаздывания, добавляемый к записям регистра режима чипом RDIMM/LRDIMM. Также следует отметить что, если используется LRDIMM, то минимальное значение этого поля - tMRD_L2 если контроллер работает в режиме соотношения частот 1:1, или tMRD_L2/2 (округленное вверх до следующего целого) если контроллер работает в режиме соотношения частот 1:2. Единицы измерения: такты синхросигнала DFI. Режим программирования: квазидинамические группы 2 и 4.		
[11:10]	–	Зарезервировано.	0	
[17:12]	t_mrd	tMRD: количество тактов, которое следует прождать после записи или чтения регистра режима. В зависимости от подсоединенной SDRAM, tMRD отражает: DDR2/mDDR: время от MRS до любой команды; DDR3/4: время от MRS до команды MRS; LPDDR2: не используется; LPDDR3/4: время от MRS до команды, не относящейся к MRS. Когда контроллер работает в режиме соотношения частот 1:2, следует запрограммировать в это поле (tMRD/2) и округлить вверх до следующего целого числа. Если используется контроль четности для DDR4, в это поле следует установить tMRD_PAR(tMOD+PL). Если подключен режим CAL (DFITMG1.dfi_t_cmd_lat > 0), следует добавлять время tCAL (=DFITMG1.dfi_cmd_lat) к вычислениям выше. Единицы измерения: такты синхросигнала DFI. Режим программирования: квазидинамические группы 2 и 4.	0x4	R/W
[19:18]	–	Зарезервировано.	0	
[29:20]	t_mrw	Время, которое следует прождать после записи или чтения регистра режима (MRW или MRR). Присутствует только в реализациях, сконфигурированных для поддержки LPDDR2, LPDDR3 или LPDDR4. LPDDR2, как правило, требует значения 5.	0x5	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		LPDDR3, как правило, требует значения 10. LPDDR4: следует устанавливать в это поле наибольшее из чисел tMRW и tMRWCKEL. Для LPDDR2, это поле регистра используется для определения времени от MRW/MRR до всех остальных команд. Когда контроллер работает в режиме соотношения частот 1:2, следует запрограммировать в это поле вышеописанные значения, деленные пополам, и округлить их вверх до следующего целого числа. Для LDPDR3, этот регистр используется для определения времени от MRW/MRR до MRW/MRR. Единицы измерения: такты синхросигнала DFI. Режим программирования: квазидинамические группы 2 и 4.		
[31:30]	–	Зарезервировано.	0	

24.6.55 Регистр задержек SDRAM 4 (DRAMTMG4)

Поля регистра DRAMTMG4 описаны в Таблица 24.57.

Таблица 24.57. Регистр задержек SDRAM 4

Диапазон	Название	Описание	Исходное значение	Тип доступа
[4:0]	t_rp	tRP: минимальное время от предзаряда до активации одного и того же банка. Когда контроллер работает в режиме соотношения частот 1:1, следует устанавливать в t_rp RoundUp(tRP/tCK). Когда контроллер работает в режиме соотношения частот 1:2, следует устанавливать в t_rp RoundDown(RoundUp(tRP/tCK)/2) + 1. Когда контроллер работает в режиме соотношения частот 1:2 в LPDDR4, следует устанавливать в t_rp RoundUp(RoundUp(tRP/tCK)/2). Единицы измерения: такты синхросигнала DFI. Режим программирования: квазидинамические группы 2 и 4.	0x5	R/W
[7:5]	–	Зарезервировано.	0	
[11:8]	t_rrd	DDR4: tRRD_L: минимальное время между активациями от банка "a" к банку "b" для одной и той же банковой группы. Другие: tRRD: минимальное время между активациями от банка "a" к банку "b". Когда контроллер работает в режиме соотношения частот 1:2, следует запрограммировать в это поле (tRRD_L/2 или tRRD/2) и округлить полученное значение вверх до следующего целого числа. Единицы измерения: такты синхросигнала DFI.	0x4	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Режим программирования: квазидинамические группы 2 и 4.		
[15:12]	–	Зарезервировано.	0	
[19:16]	t_ccd	DDR4: tCCD_L: минимальное время между двумя чтениями или двумя записями для одной и той же группы банков. Другие: tCCD: минимальное время между двумя чтениями или двумя записями. Когда контроллер работает в режиме соотношения частот 1:2, следует запрограммировать в это поле (tCCD_L/2 или tCCD/2) и округлить полученное значение вверх до следующего целого числа. Единицы измерения: такты синхросигнала DFI. Режим программирования: квазидинамические группы 2 и 4.	0x4	R/W
[23:20]	–	Зарезервировано.	0	
[28:24]	t_rcd	tRCD - tAL: минимальное время от активации до команды записи или чтения к одному и тому же банку. Когда контроллер работает в режиме соотношения частот 1:2, следует запрограммировать в это поле ((tRCD - tAL)/2) и округлить полученное значение вверх до следующего целого числа. Минимальное значение, разрешенное для этого поля регистра - 1, из чего следует, что минимальное значение (tRCD - tAL) будет равным 2, когда контроллер работает в режиме соотношения частот 1:2. Единицы измерения: такты синхросигнала DFI. Режим программирования: квазидинамические группы 1, 2 и 4.	0x5	R/W
[31:29]	–	Зарезервировано.	0	

24.6.56 Регистр задержек SDRAM 5 (DRAMTMG5)

Поля регистра DRAMTMG5 описаны в Таблица 24.58.

Таблица 24.58. Регистр задержек SDRAM 5

Диапазон	Название	Описание	Исходное значение	Тип доступа
[4:0]	t_cke	Минимальное количество тактов CKE HIGH/LOW во время понижения питания (power-down) и саморегенерации (self refresh). Режим LPDDR2/LPDDR3: устанавливает в это поле наибольшее из значений tCKE или tCKESR. Режим LPDDR4: устанавливает в это поле наибольшее из значений tCKE или tSR. Реализации, не относящиеся к LPDDR2/LPDDR3/LPDDR4: устанавливает в это поле значение tCKE.	0x3	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Когда контроллер работает в режиме соотношения частот 1:2, следует запрограммировать в это поле (вышеописанное значение)/2 и округлить его вверх до следующего целого числа. Единицы измерения: такты синхросигнала DFI. Режим программирования: квазидинамические группы 2 и 4.		
[7:5]	–	Зарезервировано.	0	
[13:8]	t_ckesr	Задержка от входа до выхода в состояние саморегенерации или саморегенерации при низком питании с минимальной шириной низкого сигнала CKE (low) в тактах синхросигнала памяти. Рекомендованные настройки: mDDR: tRFC; LPDDR2: tCKESR; LPDDR3: tCKESR; LPDDR4: max(tCKE, tSR); DDR2: tCKE; DDR3: tCKE + 1; DDR4: tCKE + 1 (+ PL(запаздывание контроля четности)(*)). (*) только если CRCPARCTL1.caparity_disable_before_sr=0, значение этого поля должно быть увеличено на PL. Когда контроллер работает в режиме соотношения частот 1:2, следует запрограммировать в это поле рекомендованное значение, деленное на 2 и округленное вверх до следующего целого числа. Единицы измерения: такты синхросигнала DFI. Режим программирования: квазидинамические группы 2 и 4.	0x4	R/W
[15:14]	–	Зарезервировано.	0	
[19:16]	t_cksre	Время после входа в состояние саморегенерации при низком питании, в течение которого сигнал CK поддерживается как валидный синхросигнал. Определяет задержку отключения синхросигнала после SRE (Self-Refresh Entry, входа в состояние саморегенерации). Рекомендованные настройки: mDDR: 0; LPDDR2: 2; LPDDR3: 2; LPDDR4: tCKELCK; DDR2: 1; DDR3: max (10 ns, 5 tCK); DDR4: max (10 ns, 5 tCK) (+ PL(запаздывание контроля четности)(*)). (*)только если CRCPARCTL1.caparity_disable_before_sr=0, значение этого поля должно быть увеличено на PL. Когда контроллер работает в режиме соотношения частот 1:2, следует запрограммировать в это поле	0x5	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		рекомендованное значение, деленное на 2 и округлить вверх до следующего целого числа. Единицы измерения: такты синхросигнала DFI. Режим программирования: квазидинамические группы 2 и 4.		
[23:20]	–	Зарезервировано.	0	
[27:24]	t_cksrx	Это время перед выходом из режима саморегенерации, в течение которого CK поддерживается как валидный синхросигнал перед отправкой SRX. Задает время стабильности синхросигнала перед SRX. Рекомендованные настройки: mDDR: 1; LPDDR2: 2; LPDDR3: 2; LPDDR4: tCKCKEH; DDR2: 1; DDR3: tCKSRX; DDR4: tCKSRX. Когда контроллер работает в режиме соотношения частот 1:2, следует запрограммировать в это поле рекомендованное значение, деленное на 2 и округлить вверх до следующего целого числа. Единицы измерения: такты синхросигнала DFI. Режим программирования: квазидинамические группы 2 и 4.	0x5	R/W
[31:28]	–	Зарезервировано.	0	

24.6.57 Регистр задержек SDRAM 6 (DRAMTMG6)

Поля регистра DRAMTMG6 описаны в Таблица 24.59.

Таблица 24.59. Регистр задержек SDRAM 6

Диапазон	Название	Описание	Исходное значение	Тип доступа
[3:0]	t_ckcsx	Это время перед выходом из остановки синхросигнала (Clock Stop Exit), в течение которого CK поддерживается как валидный синхросигнал перед отправкой команды выхода из остановки синхросигнала. Задает время, в течение которого синхросигнал стабилен перед следующей командой после выхода из остановки синхросигнала. Рекомендованные настройки: mDDR: 1; LPDDR2: tXP + 2; LPDDR3: tXP + 2; LPDDR4: tXP + 2. Когда контроллер работает в режиме соотношения частот 1:2, следует запрограммировать в это поле рекомендованное значение, деленное на 2 и округленное вверх до следующего целого числа.	0x5	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Присутствует только для реализаций, поддерживающих устройства с mDDR или с LPDDR2/LPDDR3/LPDDR4. Единицы измерения: такты синхросигнала DFI. Режим программирования: квазидинамические группы 2 и 4.		
[15:4]	–	Зарезервировано.	0	
[19:16]	t_ckdpdx	Время перед выходом из глубокого режима низкого питания (Deep Power Down Exit), в течение которого CK поддерживается как валидный синхросигнал перед отправкой DPDX (Deep Power Down Exit). Определяет время, в течение которого синхросигнал стабилен перед DPDX. Рекомендованные настройки: mDDR: 1; LPDDR2: 2; LPDDR3: 2. Когда контроллер работает в режиме соотношения частот 1:2, следует запрограммировать в это поле рекомендованное значение, деленное на 2 и округленное вверх до следующего целого числа. Присутствует только для реализаций, поддерживающих устройства с mDDR или с LPDDR2. Единицы измерения: такты синхросигнала DFI. Режим программирования: квазидинамические группы 2 и 4.	0x2	R/W
[23:20]	–	Зарезервировано.	0	
[27:24]	t_ckdpde	Время после входа в глубокий режим низкого питания (Deep Power Down Entry), в течение которого CK поддерживается как валидный синхросигнал. Определяет задержку отключения синхросигнала после DPDE (Deep Power Down Entry). Рекомендованные настройки: mDDR: 0; LPDDR2: 2; LPDDR3: 2. Когда контроллер работает в режиме соотношения частот 1:2, следует запрограммировать в это поле рекомендованное значение, деленное на 2 и округленное вверх до следующего целого числа. Присутствует только для реализаций, поддерживающих устройтсва с mDDR или с LPDDR2/LPDDR3. Единицы измерения: такты синхросигнала DFI. Режим программирования: квазидинамические группы 2 и 4.	0x2	R/W
[31:28]	–	Зарезервировано.	0	

24.6.58 Регистр задержек SDRAM 7 (DRAMTMG7)

Поля регистра DRAMTMG7 описаны в Таблица 24.60.

Таблица 24.60. Регистр задержек SDRAM 7

Диапазон	Название	Описание	Исходное значение	Тип доступа
[3:0]	t_ckpdx	Время перед выходом из режима низкого питания (Power Down Exit), в течение которого CK поддерживается как валидный синхросигнал перед отправкой PDX (Power Down Exit). Определяет время, в течение которого синхросигнал стабилен перед PDX. Рекомендованные настройки: mDDR: 0; LPDDR2: 2; LPDDR3: 2; LPDDR4: tCKCKEH. Когда используется DDR2/3/4 SDRAM, в это поле следует устанавливать то же значение, что и в DRAMTMG5.t_cksrx. Когда контроллер работает в режиме соотношения частот 1:2, следует запрограммировать в это поле рекомендованное значение, деленное на 2 и округленное вверх до следующего целого числа. Присутствует только для реализаций, поддерживающих устройства с mDDR или с LPDDR2/LPDDR3/LPDDR4. Единицы измерения: такты синхросигнала DFI. Режим программирования: квазидинамические группы 2 и 4.	0x2	R/W
[7:4]	–	Зарезервировано.	0	
[11:8]	t_ckpde	Время после входа в режим низкого питания (Power Down Entry), в течение которого CK поддерживается как валидный синхросигнал. Определяет задержку отключения синхросигнала после PDE (Power Down Entry). Рекомендованные настройки: mDDR: 0; LPDDR2: 2; LPDDR3: 2; LPDDR4: tCKELCK. Когда используется DDR2/3/4 SDRAM, в это поле следует устанавливать то же самое значение, что и в DRAMTMG5.t_cksre. Когда контроллер работает в режиме соотношения частот 1:2, следует запрограммировать в это поле рекомендованное значение, деленное на 2 и округленное вверх до следующего целого числа. Присутствует только для реализаций, поддерживающих устройства с mDDR или с LPDDR2/LPDDR3/LPDDR4. Единицы измерения: такты синхросигнала DFI. Режим программирования: квазидинамические группы 2 и 4.	0x2	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
[31:12]	–	Зарезервировано.	0	

24.6.59 Регистр задержек SDRAM 8 (DRAMTMG8)

Поля регистра DRAMTMG8 описаны в Таблица 24.61.

Таблица 24.61. Регистр задержек SDRAM 8

Диапазон	Название	Описание	Исходное значение	Тип доступа
[6:0]	t_xs_x32	tXS: время от выхода из саморегенерации (Exit Self Refresh) до получения команд, не требующих замкнутого DLL. Когда контроллер работает в режиме соотношения частот 1:2, следует запрограммировать в это поле вышеописанное значение, деленное пополам и округленное вверх до следующего целого числа. Замечание: используется только в DDR2, DDR3 и DDR4 SDRAMs. Единицы измерения: числа, кратные 32 тактам синхросигнала DFI. Режим программирования: квазидинамические группы 2 и 4.	0x5	R/W
[7]	–	Зарезервировано.	0	
[14:8]	t_xs_dll_x32	tXSDLL: время от выхода из саморегенерации (Exit Self Refresh) до получения команд, для которых требуется замкнутый DLL. Когда контроллер работает в режиме соотношения частот 1:2, следует запрограммировать в это поле вышеописанное значение, деленное на 2 и округленное вверх до следующего целого числа. Замечание: используется только в DDR2, DDR3 и DDR4 SDRAMs. Единицы измерения: числа, кратные 32 тактам синхросигнала DFI. Режим программирования: квазидинамические группы 2 и 4.	0x44	R/W
[15]	–	Зарезервировано.	0	
[22:16]	t_xs_abort_x32	tXS_ABORT: время от выхода из режима саморегенерации (Exit Self Refresh) до получения команд, не требующих замкнутого DLL в состоянии преждевременного прекращения саморегенерации (Self Refresh Abort). Когда контроллер работает в режиме соотношения частот 1:2, следует запрограммировать в это поле вышеописанное значение, деленное на 2 и округленное вверх до следующего целого числа. Замечание: следует убедиться, что значение этого поля меньше или равно t_xs_x32. Единицы измерения: числа, кратные 32 тактам синхросигнала DFI.	0x3	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Режим программирования: квазидинамические группы 2 и 4.		
[23]	–	Зарезервировано.	0	
[30:24]	t_xs_fast_x32	tXS_FAST: время от выхода из режима саморегенерации (Exit Self Refresh) в состояния ZQCL, ZQCS и MRS (только в режимах CL, WR, RTP и замедления (Geardown)). Когда контроллер работает в режиме соотношения частот 1:2, следует запрограммировать в это поле вышеописанное значение, деленное на 2 и округленное вверх до следующего целого числа. Замечание: это применимо только к командам ZQCL/ZQCS. Замечание: следует убедиться, что значение этого поля меньше или равно значению t_xs_x32. Единицы измерения: числа, кратные 32 тактам синхросигнала DFI. Режим программирования: квазидинамические группы 2 и 4.	0x3	R/W
[31]	–	Зарезервировано.	0	

24.6.60 Регистр задержек SDRAM 9 (DRAMTMG9)

Поля регистра DRAMTMG9 описаны в Таблица 24.62.

Таблица 24.62. Регистр задержек SDRAM 9

Диапазон	Название	Описание	Исходное значение	Тип доступа
[5:0]	wr2rd_s	CWL + PL + BL/2 + tWTR_S. Минимальное время от команды записи до команды чтения для разных групп банков. Включает время на обращение шины, времена восстановления, и все по-банковые, по-ранговые и глобальные ограничения. Присутствует только в реализациях, сконфигурированных для поддержки DDR4. В формуле: CWL = запаздывание записи CAS; PL = запаздывание контроля четности (parity latency); BL = длина импульса (burst length). Это значение должно соответствовать значению, запрограммированному в бит BL регистра режима к SDRAM. tWTR_S = внутренняя задержка от команды записи до команды чтения для разных групп банков. Это значение извлекается напрямую из спецификации SDRAM. WTR_S должно быть увеличено на 1 если используется начальная часть записи (начальная часть записи) DDR4 2tCK. Когда контроллер работает в режиме соотношения частот 1:2, следует разделить вычисленное выше	0xd	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		значение на 2 и округлить его вверх до следующего целого числа. Единицы измерения: такты синхросигнала DFI. Режим программирования: квазидинамические группы 1, 2 и 4.		
[7:6]	–	Зарезервировано.	0	
[11:8]	t_rrd_s	tRRD_S: минимальное время между активациями от банка "a" до банка "b" для разных банковых групп. Когда контроллер работает в режиме соотношения частот 1:2, следует запрограммировать в это поле (tRRD_S/2) и округлить полученное значение вверх до следующего целого числа. Присутствует только в реализациях, сконфигурированных для поддержки DDR4. Единицы измерения: такты синхросигнала DFI. Режим программирования: квазидинамические группы 2 и 4.	0x4	R/W
[15:12]	–	Зарезервировано.	0	
[18:16]	t_ccd_s	tCCD_S: минимальное время между двумя чтениями или двумя записями для разных групп банков. Для переключения банков (от банка "a" до банка "b"), минимальное время равно значению в этом поле + 1. Когда контроллер работает в режиме соотношения частот 1:2, следует запрограммировать в это поле (tCCD_S/2) и округлить полученное значение вверх до следующего целого числа. Присутствует только в реализациях, сконфигурированных для поддержки DDR4. Единицы измерения: такты синхросигнала DFI. Режим программирования: квазидинамические группы 2 и 4.	0x4	R/W
[29:19]	–	Зарезервировано.	0	
[30]	ddr4_wr_preamble	Режим начальной части записи (Write preamble mode) DDR4. 0: начальная часть записи равна 1tCK; 1: начальная часть записи равна 2tCK. Присутствует только когда MEMC_FREQ_RATIO=2. Режим программирования: квазидинамические группы 2 и 4.	0x0	R/W
[31]	–	Зарезервировано.	0	

24.6.61 Регистр задержек SDRAM 11 (DRAMTMG11)

Поля регистра DRAMTMG11 описаны в Таблица 24.63.

Таблица 24.63. Регистр задержек SDRAM 11

Диапазон	Название	Описание	Исходное значение	Тип доступа
[4:0]	t_ckmpe	tCKMPE: минимальное требуемое время поддержания валидного синхросигнала после входа в	0x1c	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		MPSM (Maximum Power Saving Mode, режим максимального энергосбережения). Присутствует только в реализациях, сконфигурированных для поддержки DDR4. Когда контроллер работает в режиме соотношения частот 1:2, следует делить вычисленное выше значение на 2 и округлять вверх до следующего целого числа. Единицы измерения: такты синхросигнала DFI. Режим программирования: квазидинамические группы 2 и 4.		
[7:5]	–	Зарезервировано.	0	
[9:8]	t_mpx_s	tMPX_S: минимальное время от запуска CS до CKE. Когда контроллер работает в режиме соотношения частот 1:2, следует запрограммировать в это поле (tMPX_S/2) и округлить полученное значение вверх до следующего целого числа. Присутствует только в реализациях, сконфигурированных для поддержки DDR4. Единицы измерения: такты синхросигнала DFI. Режим программирования: квазидинамические группы 2 и 4.	0x2	R/W
[15:10]	–	Зарезервировано.	0	
[20:16]	t_mpx_lh	tMPX_LH: минимальное время удержания низкого сигнала CS_n Low до переднего фронта сигнала CKE. Когда контроллер работает в режиме соотношения частот 1:2, следует запрограммировать в это поле RoundUp(tMPX_LH/2)+1. Присутствует только в реализациях, сконфигурированных для поддержки DDR4. Единицы измерения: такты синхросигнала DFI. Режим программирования: квазидинамические группы 2 и 4.	0xc	R/W
[23:21]	–	Зарезервировано.	0	
[30:24]	post_mpsm_gap_x32	tXMPDLL: минимальное время от выхода из MPSM (Maximum Power Saving Mode) до команд, требующих замкнутого DLL. Когда контроллер работает в режиме соотношения частот 1:2, следует запрограммировать в это поле (tXMPDLL/2) и округлить полученное значение вверх до следующего целого числа. Присутствует только в реализациях, сконфигурированных для поддержки DDR4. Единицы измерения: числа, кратные 32 тактам синхросигнала DFI. Режим программирования: квазидинамические группы 2 и 4.	0x44	R/W
[31]	–	Зарезервировано.	0	

24.6.62 Регистр задержек SDRAM 12 (DRAMTMG12)

Поля регистра DRAMTMG12 описаны в Таблица 24.64.

Таблица 24.64. Регистр задержек SDRAM 12

Диапазон	Название	Описание	Исходное значение	Тип доступа
[4:0]	t_mrd_pda	tMRD_PDA: время цикла установки команды регистром режима (Mode Register Set command cycle) в режиме PDA. Когда контроллер работает в режиме соотношения частот 1:2, следует запрограммировать в это поле (tMRD_PDA/2) и округлить полученное значение вверх до следующего целого числа. Единицы измерения: такты синхросигнала DFI. Режим программирования: квазидинамические группы 2 и 4.	0x10	R/W
[15:5]	–	Зарезервировано.	0	
[17:16]	t_cmdcke	tCMDCKE: задержка от валидной команды до низкого значения на вводе сигнала CKE (CKE input LOW). В это поле следует устанавливать наибольшее значение из tESCKE или tCMDCKE. Когда контроллер работает в режиме соотношения частот 1:2, следует запрограммировать в это поле (max(tESCKE, tCMDCKE)/2) и округлить его вверх до следующего целого числа. Единицы измерения: такты синхросигнала DFI. Режим программирования: квазидинамические группы 2 и 4.	0x2	R/W
[31:18]	–	Зарезервировано.	0	

24.6.63 Регистр задержек SDRAM 13 (DRAMTMG13)

Поля регистра DRAMTMG13 описаны в Таблица 24.65.

Таблица 24.65. Регистр задержек SDRAM 13

Диапазон	Название	Описание	Исходное значение	Тип доступа
[2:0]	t_ppd	LPDDR4: tPPD: это минимальное время от предзаряда до команды предзаряда. Когда контроллер работает в режиме соотношения частот 1:2, следует запрограммировать в это поле (tPPD/2) и округлить его вверх до следующего целого числа. Единицы измерения: такты синхросигнала DFI. Режим программирования: квазидинамические группы 2 и 4.	0x4	R/W
[15:3]	–	Зарезервировано.	0	
[21:16]	t_ccd_mw	LPDDR4: tCCDMW: минимальное время от записи (или маскированной записи, masked write) до команды маскированной записи для одного и того же банка. Когда контроллер работает в режиме соотношения частот 1:2, следует запрограммировать в это поле (tCCDMW/2) и округлить полученное значение вверх до следующего целого числа.	0x20	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Единицы измерения: такты синхросигнала DFI. Режим программирования: квазидинамические группы 2 и 4.		
[23:22]	–	Зарезервировано.	0	
[30:24]	odtloff	LPDDR4: tODTLoff: запаздывание от команды CAS-2 до указателя на tODToff (tODToff reference). Когда контроллер работает в режиме соотношения частот 1:2, следует запрограммировать в это поле (tODTLoff/2) и округлить полученное значение вверх до следующего целого числа. Единицы измерения: такты синхросигнала DFI. Режим программирования: квазидинамические группы 2 и 4.	0x1c	R/W
[31]	–	Зарезервировано.	0	

24.6.64 Регистр задержек SDRAM 14 (DRAMTMG14)

Поля регистра DRAMTMG14 описаны в Таблица 24.66.

Таблица 24.66. Регистр задержек SDRAM 14

Диапазон	Название	Описание	Исходное значение	Тип доступа
[11:0]	t_xsr	tXSR: время от выхода из саморегенерации (Exit Self Refresh) до любой команды. Когда контроллер работает в режиме соотношения частот 1:2, следует запрограммировать в это поле вышеописанное значение, деленное на 2 и округленное вверх до следующего целого числа. Замечание: используется только в режимах mDDR/LPDDR2/LPDDR3/LPDDR4. Единицы измерения: такты синхросигнала DFI. Режим программирования: квазидинамические группы 2 и 4.	0xa0	R/W
[31:12]	–	Зарезервировано.	0	

24.6.65 Регистр задержек SDRAM 15 (DRAMTMG15)

Поля регистра DRAMTMG15 описаны в Таблица 24.67.

Таблица 24.67. Регистр задержек SDRAM 15

Диапазон	Название	Описание	Исходное значение	Тип доступа
[7:0]	t_stab_x32	tSTAB: время стабилизации. Требуется в следующих двух случаях для DDR3/DDR4 RDIMM: - при выходе из режима энергосбережения, если синхросигнал был остановлен. После переподклчения синхросигнал должен быть стабилен в течение времени, определенного полем tSTAB;	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		- в случае изменения частоты синхросигнала на вводе (DDR4); - после отправки контрольных слов (control words) которые относятся к задержкам синхросигналов (в спецификации: 6мкс для DDR3, 5мкс для DDR4); Когда контроллер работает в режиме соотношения частот 1:2, следует запрограммировать в это поле рекомендованное значение, деленное на 2 и округленное вверх до следующего целого числа. Единицы измерения: числа, кратные 32 тактам синхросигнала DFI. Режим программирования: квазидинамические группы 2 и 4.		
[30:8]	–	Зарезервировано.	0	
[31]	en_dfi_lp_t_stab	1 - подключает использование задержки tSTAB после выхода из режима низкого питания DFI (LP). Необходимо, чтобы это поле было установлено, когда PHY останавливает синхросигнал в процессе режима низкого питания DFI (DFI LP) для максимального энергосбережения. 0 - отключает использование задержки tSTAB при выходе из DFI LP. Режим программирования: квазидинамические группы 2 и 4.	0x0	R/W

24.6.66 Регистр управления ZQ 0 (ZQCTL0)

Поля регистра ZQCTL0 описаны в Таблица 24.68.

Таблица 24.68. Регистр управления ZQ 0

Диапазон	Название	Описание	Исходное значение	Тип доступа
[9:0]	t_zq_short_nop	tZQCS для DDR3/DD4/LPDDR2/LPDDR3, tZQLAT для LPDDR4: число тактов синхросигнала DFI для команды NOP, которая требуется после отправки к SDRAM команды ZQCS (короткой калибровки ZQ)/MPC(ZQ-защёлки). Когда контроллер работает в режиме соотношения частот 1:2, следует запрограммировать в это поле tZQCS/2 и округлить вверх то следующего целого числа. Присутствует только для реализаций, поддерживающих устройства с DDR3/DDR4 или с LPDDR2/LPDDR3/LPDDR4. Единицы измерения: такты синхросигнала DFI. Режим программирования: статический.	0x40	R/W
[15:10]	–	Зарезервировано.	0	
[26:16]	t_zq_long_nop	tZQoper для DDR3/DDR4, tZQCL для LPDDR2/LPDDR3, tZQCAL для LPDDR4: число тактов синхросигнала DFI для команды NOP, которая требуется после отправки к SDRAM	0x200	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		команды ZQCL (длинной калибровки ZQ)/MPC(Старт ZQ). Когда контроллер работает в режиме соотношения частот 1:2: DDR3/DDR4: следует запрограммировать в это поле tZQoper/2 и округлить его вверх до следующего целого числа. LPDDR2/LPDDR3: следует запрограммировать в это поле tZQCL/2 и округлить его вверх до следующего целого числа. LPDDR4: следует запрограммировать в это поле tZQCAL/2 и округлить полученное значение вверх до следующего целого числа. Присутствует только для реализаций, поддерживающих устройства DDR3/DDR4 или LPDDR2/LPDDR3/LPDDR4. Единицы измерения: такты синхросигнала DFI. Режим программирования: статический.		
[27]	–	Зарезервировано.	0	
[28]	dis_mpsmx_zqcl	1 - отключает отправку команды ZQCL при выходе из режима максимального энергосбережения (Maximum Power Saving Mode, MPSM). Применимо только при работе в режиме DDR4. 0 - подключает отправку команды ZQCL при выходе из режима максимального энергосбережения. Применимо только при работе в режиме DDR4. Присутствует только для реализаций, поддерживающих устройства с DDR4. Замечание: не следует производить отправку команды ZQCL при выходе из режима максимального энергосбережения, если установлен параметр конфигурации UMCTL2_SHARED_AC. Следует программировать в это поле 1'b1. Программное обеспечение может произвести отправку ZQCS после выхода из режима MPSM. Режим программирования: статический.	0x0	R/W
[29]	zq_resistor_shared	1 - указывает, что резистор ZQ разделен между рангами. Это значит, что команды ZQinit/ZQCL/ZQCS/MPC(калибровка ZQ) посылаются к одному рангу за раз в соответствии с задержками tZQinit/tZQCL/tZQCS/tZQCAL/tZQLAT между командами, так чтобы команды к разным рангам не накладывались друг на друга. 0 - резистор ZQ не разделен. Присутствует только для реализаций, поддерживающих устройства с DDR3/DDR4 или с LPDDR2/LPDDR3/LPDDR4. Режим программирования: статический.	0x0	R/W
[30]	dis_srx_zqcl	1 - отключает отправку команды ZQCL/MPC(калибровка ZQ) при выходе из саморегенерации/саморегенерации при низком питании (SR-Powerdown). Применимо только при	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		запуске в режиме DDR3 или DDR4 или LPDDR2 или LPDDR3 или LPDDR4. 0 - подключает отправку команды ZQCL/MPC(калибровка ZQ) при выходе из саморегенерации/саморегенерации при низком питании (SR-Powerdown). Применимо только при запуске в режиме DDR3 или DDR4 или LPDDR2 или LPDDR3 или LPDDR4. Присутствует только для реализаций, поддерживающих устройства с DDR3/DDR4 или с LPDDR2/LPDDR3/LPDDR4. Режим программирования: квазидинамические группы 2 и 4.		
[31]	dis_auto_zq	1 - отключает генерацию через uMCTL2 команд ZQCS/MPC(калибровка ZQ). Вместо этого может быть использовано поле DBGCMD.zq_calib_short для отправки запроса от модуля APB на калибровку ZQ. 0 - внутренняя генерация команд ZQCS/MPC(калибровка ZQ) на основе ZQCTL1.t_zq_short_interval_x1024. Присутствует только для реализаций, поддерживающих устройства с DDR3/DDR4 или с LPDDR2/LPDDR3/LPDDR4. Режим программирования: динамический.	0x0	R/W

24.6.67 Регистр управления ZQ 1 (ZQCTL1)

Поля регистра ZQCTL1 описаны в Таблица 24.69.

Таблица 24.69. Регистр управления ZQ 1

Диапазон	Название	Описание	Исходное значение	Тип доступа
[19:0]	t_zq_short_interval_x1024	Средний интервал ожидания между автоматической отправкой команд ZQCS (короткой калибровки ZQ)/MPC (калибровки ZQ) к устройствам DDR3/DDR4/LPDDR2/LPDDR3/LPDDR4. Не имеет смысла, если ZQCTL0.dis_auto_zq=1. Это поле присутствует только для реализаций, поддерживающих устройства DDR3/DDR4 или LPDDR2/LPDDR3/LPDDR4. Единицы измерения: числа, кратные 1024 тактам синхросигнала DFI. Режим программирования: статический.	0x100	R/W
[29:20]	t_zq_reset_nop	tZQReset: количество тактов синхросигнала DFI для NOP (команд "No Operation"), требуемых после отправки команды ZQReset (сброса калибровки ZQ) к SDRAM.	0x20	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Когда контроллер работает в режиме соотношения частот 1:2, следует запрограммировать в это поле tZQReset/2 и округлить вверх до следующего целого значения. Это поле присутствует только для реализаций, поддерживающих устройства LPDDR2/LPDDR3/LPDDR4. Единицы измерения: такты синхросигнала DFI. Режим программирования: статический.		
[31:30]	–	Зарезервировано.	0	

24.6.68 Регистр управления ZQ 2 (ZQCTL2)

Поля регистра ZQCTL2 описаны в Таблица 24.70.

Таблица 24.70. Регистр управления ZQ 2

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	zq_reset	Установка 1 в это поле запускает операцию сброса ZQ (ZQ Reset). Когда операция сброса ZQ завершена, uMCTL2 очищает этот бит автоматически. Рекомендуется НЕ устанавливать этот бит во время пребывания в режимах инициализации, саморегенерации (кроме LPDDR4), саморегенерации с понижением питания SR-Powerdown (LPDDR4) или глубокого понижения питания Deep power-down. Для режима саморегенерации (кроме LPDDR4) или SR-Powerdown(LPDDR4) операция сброса ZQ будет добавлена в расписание после выхода из SR(кроме LPDDR4) или SRPD(LPDDR4). Для глубокого режима низкого питания она не будет добавлена в расписание, хотя ZQSTAT.zq_reset_busy будет снят. Это поле присутствует только для реализаций, поддерживающих устройства LPDDR2/LPDDR3/LPDDR4. Контролепригодность: readOnly. Режим программирования: динамический.	0x0	R/W
[31:1]	–	Зарезервировано.	0	

24.6.69 Регистр статуса ZQ (ZQSTAT)

Поля регистра ZQSTAT описаны в Таблица 24.71.

Таблица 24.71. Регистр статуса ZQ

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	zq_reset_busy	Система SoC может инициировать операцию сброса ZQ только если этот сигнал имеет низкое значение. Этот сигнал получает высокое значение в синхросигнале после того, как uMCTL2 принимает запрос о сбросе ZQ. Он получает низкое значение, когда команда сброса ZQ отправлена к SDRAM и заканчивается период соотвествующих ей NOP. Рекомендуется не производить команд сброса ZQ, когда этот сигнал имеет высокое значение. 0: указывает, что система SoC может инициировать операцию сброса ZQ; 1: указывает, что операция сброса ZQ находится в процессе исполнения. Режим программирования: динамический.	0x0	R/W
[31:1]	–	Зарезервировано.	0	

24.6.70 Регистр задержек DFI 0 (DFITMG0)

Поля регистра DFITMG0 описаны в Таблица 24.72.

Таблица 24.72. Регистр задержек DFI 0

Диапазон	Название	Описание	Исходное значение	Тип доступа
[5:0]	dfi_tphy_wrlat	Запаздывание записи. Количество тактов от команды записи до подключения записи данных (dfi_wrdata_en). Это соответствует параметру задержек DFI tphy_wrlat. Правильное значение этого поля указано в спецификации PHY. Следует отметить что, в зависимости от PHY, если используются RDIMM/LRDIMM, может быть необходимо использовать модифицированное значение CL при вычислении tphy_wrlat. Это нужно для того, чтобы скомпенсировать лишние такты запаздывания через RDIMM/LRDIMM. Для LPDDR4, dfi_tphy_wrlat>60 не поддерживается. Единицы измерения: такты синхросигнала DFI или такты синхросигнала PHY в DFI, в зависимости от DFITMG0.dfi_wrdata_use_dfi_phy_clk. Режим программирования: квазидинамические группы 1 и 4.	0x2	R/W
[7:6]	–	Зарезервировано.	0	
[13:8]	dfi_tphy_wrdata	Определяет количество тактов синхросигнала между моментом установки dfi_wrdata_en и моментом, когда соответствующая запись данных отсылается на сигнал dfi_wrdata. Это соответствует параметру задержек DFI tphy_wrdata. Правильное значение этого поля указано в спецификации PHY. Следует отметить, что максимальное поддерживаемое значение этого поля 8.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Единицы измерения: такты синхросигнала DFI или такты синхросигнала PHY в DFI, в зависимости от DFITMG0.dfi_wrdata_use_dfi_phy_clk. Режим программирования: квазидинамическая группа 4.		
[14]	–	Зарезервировано.	0	
[15]	dfi_wrdata_use_dfi_phy_clk	Определяет, следует ли генерировать dfi_wrdata_en/dfi_wrdata/dfi_wrdata_mask с использованием значения HDR (синхросигнал DFI) или SDR (синхросигнал PHY в DFI). Позволяет выбрать, будет ли значение DFITMG0.dfi_tphy_wrlat измеряться в тактах HDR (синхросигнал DFI) или SDR (синхросигнал PHY в DFI). Позволяет выбрать, будет ли значение DFITMG0.dfi_tphy_wrdata измеряться в тактах HDR (синхросигнал DFI) или SDR (синхросигнал PHY в DFI). 0: в тактах HDR (синхросигнал DFI); 1: в тактах SDR (синхросигнал PHY в DFI). Правильное значение этого поля указано в спецификации PHY. Если используется DDR3/2 PHY, DDR2/3-Lite/mDDR PHY, DDR multiPHY или Gen2 DDR multiPHY от Synopsys DWC, в это поле должен быть установлен 0; в противном случае: если MEMC_PROG_FREQ_RATIO=1 и MSTR.frequency_ratio=1, в это поле должен быть установлен 0. В противном случае, в него следует устанавливать 1. Режим программирования: статический.	0x0	R/W
[22:16]	dfi_t_rddata_en	Время от установки команды чтения на интерфейсе DFI до установки сигнала dfi_rddata_enl. Правильное значение этого поля указано в спецификации PHY. Это поле соответствует параметру DFI trddata_en. Следует отметить что, в зависимости от PHY, если используется RDIMM/LRDIMM, может быть необходимо использовать модифицированное значение CL в расчетах trddata_en. Это нужно для того, чтобы скомпенсировать лишние такты задержки через RDIMM/LRDIMM. Единицы измерения: такты синхросигнала DFI или такты синхросигнала PHY в DFI, в зависимости от DFITMG0.dfi_rddata_use_dfi_phy_clk. Режим программирования: квазидинамические группы 1 и 4.	0x2	R/W
[23]	dfi_rddata_use_dfi_phy_clk	Определяет, следует ли генерировать dfi_rddata_en/dfi_rddata/dfi_rddata_valid с использованием значения HDR (синхросигнал DFI) или SDR (синхросигнал PHY в DFI).	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Позволяет выбрать, будет ли значение DFITMG0.dfi_t_rddata_en измеряться в тактах HDR (синхросигнал DFI) или SDR (синхросигнал PHY в DFI): 0: в тактах HDR (синхросигнал DFI); 1: в тактах SDR (синхросигнал PHY в DFI). Правильное значение этого поля указано в спецификации PHY. Если используется DDR3/2 PHY, DDR2/3-Lite/mDDR PHY, DDR multiPHY или Gen2 DDR multiPHY от Synopsys DWC, в это поле должен быть установлен 0; в противном случае: если MEMC_PROG_FREQ_RATIO=1 и MSTR.frequency_ratio=1, в это поле должен быть установлен 0. В противном случае, в него следует устанавливать 1. . Режим программирования: статический.		
[28:24]	dfi_t_ctrl_delay	Определяет количество тактов синхросигнала DFI после установки или снятия тех сигналов настройки DFI, которыми сигналы настройки в интерфейсе PHY-DRAM указывают установку или снятие. Если синхросигнал DFI и синхросигнал памяти не совпадают по фазе, этот параметр задержки должен быть округлен вверх до следующего целого значения. Следует отметить, что, если используется RDIMM/LRDIMM, необходимо инкрементировать этот параметр на лишний такт запаздывания RDIMM's/LRDIMM's в тактах синхросигнала DFI. Единицы измерения: такты синхросигнала DFI. Режим программирования: квазидинамическая группа 4.	0x7	R/W
[31:29]	–	Зарезервировано.	0	

24.6.71 Регистр задержек DFI 1 (DFITMG1)

Поля регистра DFITMG1 описаны в Таблица 24.73.

Таблица 24.73. Регистр задержек DFI 1

Диапазон	Название	Описание	Исходное значение	Тип доступа
[4:0]	dfi_t_dram_clk_enable	Определяет количество тактов синхросигнала DFI от снятия сигнала dfi_dram_clk_disable на DFI до первого валидного восходящего фронта синхросигнала к устройствам памяти DRAM на контуре PHY-DRAM (PHY-DRAM boundary). Если синхросигнал DFI и синхросигнал памяти не совпадают по фазе, этот параметр задержки следует округлить вверх до следующего целого числа. Единицы измерения: такты синхросигнала DFI. Режим программирования: квазидинамическая группа 4.	0x4	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
[7:5]	–	Зарезервировано.	0	
[12:8]	dfi_t_dram_clk_disable	Определяет количество тактов синхросигнала DFI от момента установки сигнала dfi_dram_clk_disable на DFI до момента, когда синхросигнал к устройствам памяти DRAM на контуре PHY-DRAM устанавливает низкое значение. Если синхросигнал DFI и синхросигнал памяти не совпадают по фазе, этот параметр задержки следует округлить вверх до следующего целого числа. Единицы измерения: такты синхросигнала DFI. Режим программирования: квазидинамическая группа 4.	0x4	R/W
[15:13]	–	Зарезервировано.	0	
[20:16]	dfi_t_wrdata_delay	Определяет количество тактов синхросигнала DFI между моментом, когда сигнал dfi_wrdata_en signal установлен и моментом, когда соответсвующая передача данных записи завершается на шине DRAM. Это соответствует параметру задержек DFI twrdata_delay. Правильное значение этого поля указано в спецификации PHY. Для DFI 3.0 PHY, следует устанавливать twrdata_delay, новый параметр задержки, введенный в DFI 3.0. Для DFI 2.1 PHY, следует устанавливвать tphy_wrdata + (задержки записи данных DFI к DRAM). Программируемое значение измеряется в единицах тактов DFI, а не тактов PHY. В FREQ_RATIO=2, следует делить значение от PHY на 2 и округлять вверх до следующего целого числа. Если используется DFITMG0.dfi_wrdata_use_dfi_phy_clk=1, следует добавлять 1 к этому значению. Единицы измерения: такты синхросигнала DFI. Режим программирования: квазидинамическая группа 4.	0x0	R/W
[23:21]	–	Зарезервировано.	0	
[25:24]	dfi_t_parin_lat	Определяет количество тактов синхросигнала PHY в DFI между моментом, когда установлен сигнал dfi_cs, и моментом, когда активирован соответствующий сигнал dfi_parity_in. Единицы измерения: такты синхросигнала PHY в DFI. Режим программирования: квазидинамическая группа 4.	0x0	R/W
[27:26]	–	Зарезервировано.	0	
[31:28]	dfi_t_cmd_lat	Определяет количество тактов синхросигнала PHY в DFI между моментом, когда установлен сигнал dfi_cs signal и моментом, когда активирована соответствующая команда. Это поле используется для режима CAL, в него следует устанавливать "0" или tCAL, что	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		соответствует установкам регистра режима CAL в DRAM. Когда режим CAL mode подключается через RDIMM/LRDIMM, в это поле следует устанавливать tCAL-CLA (Command Latency Adder - сумматор запаздывания команд). Более подробно CLA описан в спецификации регистров DDR4 от JEDEC. Если PHY может добавлять запаздывание для режима CAL, в это поле следует устанавливать 0. Валидный диапазон: от 0 до 8. Единицы измерения: такты синхросигнала PHY в DFI. Режим программирования: квазидинамические группы 2 и 4.		

24.6.72 Регистр конфигурирования интерфейса низкого питания DFI 0 (DFILPCFG0)

Поля регистра DFILPCFG0 описаны в Таблица 24.74.

Таблица 24.74. Регистр конфигурирования интерфейса низкого питания DFI 0

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	dfi_lp_en_pd	Подключает согласование (handshaking) интерфейса низкого питания DFI в процессе входа в/выхода из режима низкого питания (Power Down): 0: отключено; 1: подключено. Режим программирования: статический.	0x0	R/W
[3:1]	–	Зарезервировано.	0	
[7:4]	dfi_lp_wakeup_pd	Значение в тактах синхросигнала DFI, которое следует установить на сигнале dfi_lp_wakeup при входе в режим низкого питания. Задаёт время tlp_wakeup DFI: 0x0: 16 тактов; 0x1: 32 такта; 0x2: 64 такта; 0x3: 128 тактов; 0x4: 256 тактов; 0x5: 512 тактов; 0x6: 1024 тактов; 0x7: 2048 тактов; 0x8: 4096 тактов; 0x9: 8192 тактов; 0xA: 16384 тактов; 0xB: 32768 тактов; 0xC: 65536 тактов; 0xD: 131072 тактов; 0xE: 262144 тактов; 0xF: неограничено.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Единицы измерения: такты синхросигнала DFI. Режим программирования: статический.		
[8]	dfi_lp_en_sr	Подключает согласование (handshaking) интерфейса низкого питания DFI в процессе входа в/выхода из режима саморегенерации: 0: отключено; 1 : подключено. Режим программирования: статический.	0x0	R/W
[11:9]	–	Зарезервировано.	0	
[15:12]	dfi_lp_wakeup_sr	Значение в тактах синхросигнала DFI, которое следует установить на сигнале dfi_lp_wakeup при входе в режим саморегенерации. Задаёт время tlp_wakeup DFI: 0x0: 16 тактов; 0x1: 32 такта; 0x2: 64 такта; 0x3: 128 тактов; 0x4: 256 тактов; 0x5: 512 тактов; 0x6: 1024 тактов; 0x7: 2048 тактов; 0x8: 4096 тактов; 0x9: 8192 тактов; 0xA: 16384 тактов; 0xB: 32768 тактов; 0xC: 65536 тактов; 0xD: 131072 тактов; 0xE: 262144 тактов; 0xF: неограничено. Единицы измерения: такты синхросигнала DFI.. Режим программирования: статический.	0x0	R/W
[16]	dfi_lp_en_dpd	Подключает согласование (handshaking) интерфейса низкого питания DFI в процессе входа в/выхода из глубокого режима низкого питания (Power Down): 0: отключено; 1 : подключено. Это поле присутствует только для реализаций, поддерживающих устройства mDDR или LPDDR2/LPDDR3. Режим программирования: статический.	0x0	R/W
[19:17]	–	Зарезервировано.	0	
[23:20]	dfi_lp_wakeup_dpd	Значение в тактах синхросигнала DFI, которое следует установить на сигнале dfi_lp_wakeup при входе в глубокий режим низкого питания. Задаёт время tlp_wakeup DFI: 0x0: 16 тактов; 0x1: 32 такта; 0x2: 64 такта; 0x3: 128 тактов; 0x4: 256 тактов; 0x5: 512 тактов; 0x6: 1024 тактов; 0x7: 2048 тактов; 0x8: 4096 тактов;	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		0x9: 8192 тактов; 0xA: 16384 тактов; 0xB: 32768 тактов; 0xC: 65536 тактов; 0xD: 131072 тактов; 0xE: 262144 тактов; 0xF: неограничено. Это поле присутствует только для реализаций, поддерживающих устройства mDDR or LPDDR2/LPDDR3 devices. Единицы измерения: такты синхросигнала DFI. Режим программирования: статический.		
[28:24]	dfi_tlp_resp	Установка времени tlp_resp DFI в тактах синхросигнала DFI. Одно и то же значение используется для режимов низкого питания (Power Down), саморегенерации (Self Refresh), глубоко пониженного питания (Deep Power Down) и максимального энергосбережения (Maximum Power Saving). Начиная с версии спецификации DFI 2.1, рекомендуется использовать для этого поля фиксированное значение 7. Единицы измерения: такты синхросигнала DFI. Режим программирования: статический.	0x7	R/W
[31:29]	–	Зарезервировано.	0	

24.6.73 Регистр конфигурирования интерфейса низкого питания DFI 1 (DFILPCFG1)

Поля регистра DFILPCFG1 описаны в Таблица 24.75.

Таблица 24.75. Регистр конфигурирования интерфейса низкого питания DFI 1

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	dfi_lp_en_mpsm	Подключает процедуру согласования (handshaking) в процессе входа в/выхода из режима максимального энергосбережения. 0: отключен; 1 : подключен. Это поле присутствует только для реализаций, поддерживающих устройства DDR4. Режим программирования: статический.	0x0	R/W
[3:1]	–	Зарезервировано.	0	
[7:4]	dfi_lp_wakeup_mpsm	Значение в тактах синхросигнала DFI, которое следует установить на сигнале dfi_lp_wakeup при входе в режим максимального энергосбережения. Задаёт время tlp_wakeup DFI: 0x0: 16 тактов; 0x1: 32 такта; 0x2: 64 такта; 0x3: 128 тактов; 0x4: 256 тактов;	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		0x5: 512 тактов; 0x6: 1024 тактов; 0x7: 2048 тактов; 0x8: 4096 тактов; 0x9: 8192 тактов; 0xA: 16384 тактов; 0xB: 32768 тактов; 0xC: 65536 тактов; 0xD: 131072 тактов; 0xE: 262144 тактов; 0xF: неограничено. Единицы измерения: такты синхросигнала DFI. Режим программирования: статический.		
[31:8]	–	Зарезервировано.	0	

24.6.74 Регистр обновлений DFI 0 (DFIUPD0)

Поля регистра DFIUPD0 описаны в Таблица 24.76.

Таблица 24.76. Регистр обновлений DFI 0

Диапазон	Название	Описание	Исходное значение	Тип доступа
[9:0]	dfi_t_ctrlup_min	Задает минимальное число тактов синхросигнала DFI, которое следует установить сигналу dfi_ctrlupd_req. uMCTL2 ожидает от PHY ответа в пределах этого времени. Если PHY не даёт ответа, uMCTL2 снимет dfi_ctrlupd_req после того, как пройдет dfi_t_ctrlup_min + 2 такта. Наименьшее значение, которое можно присвоить этому полю, равно 0x3. Единицы измерения: такты синхросигнала DFI. Режим программирования: статический.	0x3	R/W
[15:10]	–	Зарезервировано.	0	
[25:16]	dfi_t_ctrlup_max	Задает максимальное число тактов синхросигнала DFI, которое может быть установлено сигналу dfi_ctrlupd_req. Наименьшее значение, которое можно присвоить этому полю - 0x40. Единицы измерения: такты синхросигнала DFI. Режим программирования: статический.	0x40	R/W
[28:26]	–	Зарезервировано.	0	
[29]	ctrlupd_pre_srx	Задает требования к dfi_ctrlupd_req при SRX (Self-Refresh Exit): 0 : отослать ctrlupd после SRX; 1 : отослать ctrlupd перед SRX. Если DFIUPD0.dis_auto_ctrlupd_srx=1, этот регистр никак не влияет, потому что во время SRX не будет отправки dfi_ctrlupd_req. Режим программирования: статический.	0x0	R/W
[30]	dis_auto_ctrlupd_srx	Когда это поле имеет значение "1", автоматическая генерация dfi_ctrlupd_req отключена через uMCTL2 при выходе из саморегенерации.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Когда это поле имеет значение "0", uMCTL2 отправляет dfi_ctrlupd_req перед или после выхода из саморегенерации, в зависимости от DFIUPD0.ctrlupd_pre_srx. Режим программирования: статический.		
[31]	dis_auto_ctrlupd	Когда это поле имеет значение "1", автоматическая генерация dfi_ctrlupd_req через uMCTL2 отключена. Система должна отправлять сигнал dfi_ctrlupd_req, используя регистр DBGCMD.ctrlupd. Когд это поле имеет значение "0", uMCTL2 отправляет dfi_ctrlupd_req периодически. Режим программирования: квазидинамическая группа 3.	0x0	R/W

24.6.75 Регистр обновлений DFI 1 (DFIUPD1)

Поля регистра DFIUPD1 описаны в Таблица 24.77.

Таблица 24.77. Регистр обновлений DFI 1

Диапазон	Название	Описание	Исходное значение	Тип доступа
[7:0]	dfi_t_ctrlupd_interval_max_x1024	Это поле задает максимальное время между инициированными uMCTL2 запросами обновления DFI. Этот таймер сбрасывается с каждым запросом обновления; когда таймер истекает, производится отправка dfi_ctrlupd_req и передача данных (traffic) блокируется до тех пор, пока не будет получен dfi_ctrlupd_ackx. PHY может использовать это время простоя для рекалибровки линий задержки к DLLs. Обновление контроллера DFI также используется для сброса указателей буферов FIFO в PHY в случае ошибок захвата данных. Обновления необходимы для поддержки калибрации PVT (мощности, напряжения и температуры), но частые обновления могут повлиять на исполнение. Минимальная разрешенная величина для этого поля - 1. Замечание: значение, запрограммированное в DFIUPD1.dfi_t_ctrlupd_interval_max_x1024 должно быть больше, чем DFIUPD1.dfi_t_ctrlupd_interval_min_x1024. Единицы измерения: числа, кратные 1024 тактам синхросигнала DFI. Режим программирования: статический.	0x1	R/W
[15:8]	–	Зарезервировано.	0	
[23:16]	dfi_t_ctrlupd_interval_min_x1024	Это поле задает минимальное время между инициированными uMCTL2 запросами обновления DFI (которые исполняются в те моменты, когда uMCTL2 находится в простое). Чтобы снизить частоту запросов обновления, следует установить в	0x1	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		это поле число побольше. Но это может иметь небольшое влияние на запаздывание запроса первого чтения, когда uMCTL2 находится в простое. Минимальное разрешенное значение для этого поля - 1. Единицы измерения: числа, кратные 1024 тактам синхросигнала DFI. Режим программирования: статический.		
[31:24]	–	Зарезервировано.	0	

24.6.76 Регистр обновлений DFI 2 (DFIUPD2)

Поля регистра DFIUPD2 описаны в Таблица 24.78.

Таблица 24.78. Регистр обновлений DFI 2

Диапазон	Название	Описание	Исходное значение	Тип доступа
[30:0]	–	Зарезервировано.	0	
[31]	dfi_phyupd_en	Подключает поддержку подтверждения обновлений, инициированных PHY: 0: отключена; 1: подключена. Режим программирования: статический.	0x1	R/W

24.6.77 Регистр прочих настроек DFI (DFIMISC)

Поля регистра DFIMISC описаны в Таблица 24.79.

Таблица 24.79. Регистр прочих настроек DFI

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	dfi_init_complete_en	Сигнал подключения завершения инициализации PHY (PHY initialization complete enable signal). Когда это поле установлено, сигнал dfi_init_complete может быть использован для запуска инициализации SDRAM. Режим программирования: квазидинамическая группа 3.	0x1	R/W
[1]	phy_dbi_mode	Это поле определяет, где реализован интерфейс DBI - в DDRC или в PHY. 0 - DDRC осуществляет функционал DBI. 1 - PHY осуществляет функционал DBI. Присутствует только в реализациях, сконфигурированных для поддержки DDR4 и LPDDR4. Режим программирования: статический.	0x0	R/W
[2]	dfi_data_cs_polarity	Определяет полярность сигналов dfi_wrdata_cs и dfi_rddata_cs. 0: сигналы имеют низкое значение: "active low";	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		1: сигналы имеют высокое значение: "active high". Режим программирования: статический.		
[3]	share_dfi_dram_clk_disable	Зарезервировано.	0x0	
[4]	ctl_idle_en	Подключает поддержку сигнала ctl_idle, который является не относящимся к DFI контактом, специфичным для определенных PHY от Synopsys. Подробнее о функционале ctl_idle следует смотреть в описании сигнала ctl_idle. Режим программирования: статический.	0x0	R/W
[5]	dfi_init_start	Сигнал запроса старта инициализации PHY. Когда это поле установлено, оно запускает запрос старта инициализации PHY. Режим программирования: квазидинамическая группа 3.	0x0	R/W
[6]	dis_dyn_adr_tri	Если в этот бит установлена 1, отключается специфическая для PHY функция динамического переключения между тремя состояниями (Dynamic Tristating). Эта функция специфична для определенных PHY от Synopsys. Если этот бит имеет значение 0, производится отправка специальной команды простоя IDLE на DFI, пока dfi_cs неактивен, так, чтобы PHY могла определить его для перевода команды/адресной шины в состояние Hi-Z (высокого импеданса). Этот функционал работает только в режиме соотношения частот 1:2 для DFI, вне зависимости от MSTR.en_2t_timing_mode, так что если любое из следующих ниже условий удовлетворено, отправки специальной команды IDLE на шину DFI не происходит: - MEMC_FREQ_RATIO==1; - MEMC_PROG_FREQ_RATIO=1 и MSTR.frequency_ratio=1. Специальная команда IDLE означет следующие коды, со случаем, при котором все dfi_cs имеют значение 1: - (фазы 0 и 1) dfi_ras_n=1; - (фазы 0 и 1) dfi_cas_n=1; - (фазы 0 и 1) dfi_we_n=1; - (фазы 0 и 1) dfi_bank [0]=0; - (фазы 0 и 1) dfi_act_n=1. Режим программирования: квазидинамическая группа 3.	0x1	R/W
[7]	–	Зарезервировано.	0	
[12:8]	dfi_frequency	Указывает рабочую частоту системы. Число предполагаемых частот и соответствия значений сигналов частотам синхросигнала определяет PHY. Режим программирования: квазидинамическая группа 1.	0x0	R/W
[31:13]	–	Зарезервировано.	0	

24.6.78 Регистр задержек DFI 2 (DFITMG2)

Поля регистра DFITMG2 описаны в Таблица 24.80.

Таблица 24.80. Регистр задержек DFI 2

Диапазон	Название	Описание	Исходное значение	Тип доступа
[5:0]	dfi_tphy_wrcslat	Количество тактов синхросигнала PHY в DFI между моментом, когда команда записи отсылается на интерфейс управления DFI и моментом, когда устанавливается соответсвующий сигнал dfi_wrdata_cs. Это соответствует параметру задержек DFI tphy_wrcslat. Правильное значение этого поля указано в спецификации PHY. Единицы измерения: такты синхросигнала PHY в DFI. Режим программирования: квазидинамические группы 2 и 4	0x2	R/W
[7:6]	–	Зарезервировано.	0	
[14:8]	dfi_tphy_rdcslat	Количество тактов синхросигнала PHY в DFI между моментом, когда команда чтения посылается на интерфейс управления DFI и моментом, когда устанавливается соответствующий сигнал dfi_rddata_cs. Это соответствует параметру задержек DFI tphy_rdcslat. Правильное значение этого поля указано в спецификации PHY. Единицы измерения: такты синхросигнала PHY в DFI. Режим программирования: квазидинамические группы 2 и 4.	0x2	R/W
[31:15]	–	Зарезервировано.	0	

24.6.79 Регистр статуса DFI (DFISTAT)

Поля регистра DFISTAT описаны в Таблица 24.81.

Таблица 24.81. Регистр статуса DFI

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	dfi_init_complete	Поле флага статуса, которое объявляет, когда была завершена инициализация DFI. Для получения информации о том, в какой момент завершена инициализиация, производится опрос (poll) команды инициализации DFI INIT, запущенной сигналом dfi_init_start signal, и затем опрос флага dfi_init_complete. Режим программирования: динамический.	0x0	R/W
[1]	dfi_lp_ack	Хранит значение ввода dfi_lp_ack к контроллеру. Режим программирования: динамический.	0x0	R/W
[31:2]	–	Зарезервировано.	0	

24.6.80 Регистр управления DM/DBI (DBICTL)

Поля регистра DBICTL описаны в Таблица 24.82.

Таблица 24.82. Регистр управления DM/DBI

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	dm_en	Сигнал подключения DM в DDRC. 0: DM отключен; 1: DM подключен. В этот сигнал должно быть установлено то же самое логическое значение, что и в регистр режима DRAM. DDR4: следует устанавливает в это поле то же значение, что и в бите A10 регистра MR5. Когда используются устройства x4, в это поле следует устанавливать 0. LPDDR4: следует устанавливать в это поле инвертированное значение бита MR13[5], который имеет полярность, противоположную этому сигналу. Режим программирования: статический.	0x1	R/W
[1]	wr_dbi_en	Сигнал подключение DBI записи в DDRC. 0: DBI запись отключена; 1: DBI запись подключена. В этот сигнал следует устанавливать то же значение, что и в регистр режима DRAM. DDR4: значение должно соответствовать биту A11 регистра MR5. Когда используются устройства x4, в это поле следует устанавливать 0. LPDDR4: MR3[7]. Режим программирования: квазидинамическая группа 1.	0x0	R/W
[2]	rd_dbi_en	Сигнал подключение DBI чтения в DDRC. 0: DBI чтение отключено; 1: DBI чтение подключено. В этот сигнал следует устанавливать то же значение, что и в регистре режима DRAM. DDR4: значение должно соответствовать биту A12 регистра MR5. Когда используются устройства x4, в это поле следует устанавливать 0. LPDDR4: MR3[6] Режим программирования: квазидинамическая группа 1.	0x0	R/W
[31:3]	–	Зарезервировано.	0	

24.6.81 Регистр интерфейса управления PHY в DFI (DFIPHYMSTR)

Поля регистра DFIPHYMSTR описаны в Таблица 24.83.

Таблица 24.83. Регистр интерфейса управления PHY в DFI

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	dfi_phymstr_en	Подключает интерфейс управления PHY (PHY Master Interface): 0: отключен; 1: подключен. Режим программирования: статический.	0x1	R/W
[31:1]	–	Зарезервировано.	0	

24.6.82 Регистр соответствия адресов 0 (ADDRMAP0)

Поля регистра ADDRMAP0 описаны в Таблица 24.84.

Таблица 24.84. Регистр соответствия адресов 0

Диапазон	Название	Описание	Исходное значение	Тип доступа
[4:0]	addrmap_cs_bit0	Это поле задает бит адреса интерфейса HIF, используемый как бит 0 адреса ранга. Валидный диапазон: от 0 до 29, и 31. Базовое внутреннее значение: 6. Выбранный бит адреса интерфейса HIF определяется путем прибавки базового внутреннего значения к значению этого поля. Если это поле не используется, в него следует установать 31, тогда в бит 0 адреса ранга будет установлен 0. . Режим программирования: статический.	0x0	R/W
[31:5]	–	Зарезервировано.	0	

24.6.83 Регистр соответствия адресов 1 (ADDRMAP1)

Поля регистра ADDRMAP1 описаны в Таблица 24.85.

Таблица 24.85. Регистр соответствия адресов 1

Диапазон	Название	Описание	Исходное значение	Тип доступа
[5:0]	addrmap_bank_b0	Это поле задает биты адреса интерфейса HIF, используемые как бит 0 адреса банка. Валидный диапазон: от 0 до 32, и 63. Базовое внутреннее значение: 2. Выбранный бит адреса интерфейса HIF для каждого из битов адреса банка определяется путём прибавления базового внутреннего значения к значению этого поля. Если это поле не используется, следует установить в него 63, тогда в бит 0 адреса банка будет установлен 0. Режим программирования: статический.	0x0	R/W
[7:6]	–	Зарезервировано.	0	
[13:8]	addrmap_bank_b1	Это поле задает биты адреса интерфейса HIF, используемые как бит 1 адреса банка.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Валидный диапазон: от 0 до 32, и 63. Базовое внутреннее значение: 3. Выбранный бит адреса интерфейса HIF для каждого из битов адреса банка определяется путём прибавления базового внутреннего значения к значению этого поля. Если это поле не используется, следует установить в него 63, тогда в бит 1 адреса банка будет установлен 0. Режим программирования: статический.		
[15:14]	–	Зарезервировано.	0	
[21:16]	addrmap_bank_b2	Это поле задает бит адреса интерфейса HIF, используемый как бит 2 адреса банка. Валидный диапазон: от 0 до 31, и 63. Базовое внутреннее значение: 4. Выбранный бит адреса интерфейса HIF определяется путём прибавления базового внутреннего значения к значению этого поля. Если это поле не используется, следует установить в него 63, тогда в бит 2 адреса банка будет установлен 0. Режим программирования: статический.	0x0	R/W
[31:22]	–	Зарезервировано.	0	

24.6.84 Регистр соответствия адресов 2 (ADDRMAP2)

Поля регистра ADDRMAP2 описаны в Таблица 24.86.

Таблица 24.86. Регистр соответствия адресов 2

Диапазон	Название	Описание	Исходное значение	Тип доступа
[3:0]	addrmap_col_b2	Режим полной ширины шины: это поле задает бит адреса интерфейса HIF, используемый как бит 2 адреса колонки. Режим полуширины шины: это поле задает бит адреса интерфейса HIF, используемый как бит 3 адреса колонки. Режим четверти ширины шины: это поле задает бит адреса интерфейса HIF, используемый как бит 4 адреса колонки. Валидный диапазон: от 0 до 7. Базовое внутреннее значение: 2. Выбранный бит адреса интерфейса HIF определяется путём прибавления базового внутреннего значения к значению этого поля. Следует отметить, что, если UMCTL2_INCL_ARB=1 и MEMC_BURST_LENGTH=8, требуется запрограммировать в это поле 0, если только не присутствуют следующие случаи: - в режимах половины или четверти ширины шины (MSTR.data_bus_width!=00) и - PCCFG.bl_exp_mode==1; и также: - в DDR4 ADDRMAP8.addrmap_bg_b0==0; или:	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		- в LPDDR4 ADDRMAP1.addrmap_bank_b0==0. Если UMCTL2_INCL_ARB=1 и MEMC_BURST_LENGTH=16, требуется запрограммировать в это поле 0, если только не присутствуют следующие случаи: - в режимах половины или четверти ширины шины (MSTR.data_bus_width!=00) и - PCCFG.bl_exp_mode==1; - также в DDR4 ADDRMAP8.addrmap_bg_b0==0. В противном случае, если MEMC_BURST_LENGTH=8 и подключен режим полной ширины шины (MSTR.data_bus_width==00), рекомендуется программировать 0 в этот бит, чтобы бит HIF[2] соответствовал биту 2 адреса колонки. Если MEMC_BURST_LENGTH=16 и подключен режим полной ширины шины (MSTR.data_bus_width==00), рекомендуется программировать 0 в этот бит, чтобы бит HIF[2] соответствовал биту 2 адреса колонки. Если MEMC_BURST_LENGTH=16 и подключен режим полуширины шины (MSTR.data_bus_width==01), рекомендуется программировать 0 в этот бит, чтобы бит HIF[2] соответствовал биту 3 адреса колонки. Режим программирования: статический.		
[7:4]	–	Зарезервировано.	0	
[12:8]	addrmap_col_b3	Режим полной ширины шины: это поле задает бит адреса интерфейса HIF, используемый как бит 3 адреса колонки. Режим полуширины шины: это поле задает бит адреса интерфейса HIF, используемый как бит 4 адреса колонки. Режим четверти ширины шины: это поле задает бит адреса интерфейса HIF, используемый как бит 5 адреса колонки. Валидный диапазон: от 0 до 7, и x. x здесь указывает валидное значение во встроенной (inline) конфигурации ECC. Базовое внутреннее значение: 3 Выбранный бит адреса интерфейса HIF определяется путём прибавления базового внутреннего значения к значению этого поля. Следует отметить, что, если UMCTL2_INCL_ARB=1, MEMC_BURST_LENGTH=16, ширина шины полная (MSTR.data_bus_width=00) и подключена BL16 (MSTR.burst_rdwr=1000), рекомендуется программировать в это поле 0. В режиме встроенной конфигурации ECC (MEMC_INLINE_ECC=1) и когда ECC подключен (ECCCFG0.ecc_mode>0), если MEMC_BURST_LENGTH=16, число адресов	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		колонок равно 10 и работа ведется в режиме полуширины шины, то биты колонок 8, 7 и 3 должны соотноситься с самыми высокими из трёх валидных битов адреса интерфейса HIF. В это поле регистра следует установить x. (где x = (самый высокий из валидных битов адреса HIF) - (базовое внутреннее значение) - 2). Режим программирования: статический.		
[15:13]	–	Зарезервировано.	0	
[19:16]	addrmap_col_b4	Режим полной ширины шины: это поле задает бит адреса интерфейса HIF, используемый как бит 4 адреса колонки. Режим полуширины шины: это поле задает бит адреса интерфейса HIF, используемый как бит 5 адреса колонки. Режим четверти ширины шины: это поле задает бит адреса интерфейса HIF, используемый как бит 6 адреса колонки. Валидный диапазон: от 0 до 7, и 15. Базовое внутреннее значение: 4. Выбранный бит адреса интерфейса HIF определяется путём прибавления базового внутреннего значения к значению этого поля. Если это поле не используется, следует установить в него 15, тогда в бит адреса этой колонки будет установлен 0. Режим программирования: статический.	0x0	R/W
[23:20]	–	Зарезервировано.	0	
[27:24]	addrmap_col_b5	Режим полной ширины шины: это поле задает бит адреса интерфейса HIF, используемый как бит 5 адреса колонки. Режим полуширины шины: это поле задает бит адреса интерфейса HIF, используемый как бит 6 адреса колонки. Режим четверти ширины шины: это поле задает бит адреса интерфейса HIF, используемый как бит 7 адреса колонки. Валидный диапазон: от 0 до 7, и 15. Базовое внутреннее значение: 5. Выбранный бит адреса интерфейса HIF определяется путём прибавления базового внутреннего значения к значению этого поля. Если это поле не используется, следует установить в него 15, тогда в бит адреса этой колонки будет установлен 0. Режим программирования: статический.	0x0	R/W
[31:28]	–	Зарезервировано.	0	

24.6.85 Регистр соответствия адресов 3 (ADDRMAP3)

Поля регистра ADDRMAP3 описаны в Таблица 24.87.

Таблица 24.87. Регистр соответствия адресов 3

Диапазон	Название	Описание	Исходное значение	Тип доступа
[4:0]	addrmap_col_b6	Режим полной ширины шины: это поле задает бит адреса интерфейса HIF, используемый как бит 6 адреса колонки. Режим полуширины шины: это поле задает бит адреса интерфейса HIF, используемый как бит 7 адреса колонки. Режим четверти ширины шины: это поле задает бит адреса интерфейса HIF, используемый как бит 8 адреса колонки. Валидный диапазон: от 0 до 7, x, и 31. x здесь указывает валидное значение во встроенной конфигурации ECC. Базовое внутреннее значение: 6. Выбранный бит адреса интерфейса HIF определяется путём прибавления базового внутреннего значения к значению этого поля. Если это поле не используется, следует установить в него 31, тогда в бит адреса этой колонки будет установлен 0. В режиме встроенной конфигурации ECC (MEMC_INLINE_ECC=1) и когда ECC подключен (ECCCFG0.ecc_mode>0), если MEMC_BURST_LENGTH=8, число адресов колонок равно 10 и работа ведется в режиме полуширины шины, то биты колонок 8, 7 и 6 должны соотноситься с наибольшими из трёх валидных битов адреса интерфейса HIF. В это поле регистра следует установить x. (где x = наибольший из валидных битов адреса HIF - базовое внутреннее значение - 2). Режим программирования: статический.	0x0	R/W
[7:5]	–	Зарезервировано.	0	
[12:8]	addrmap_col_b7	Режим полной ширины шины: это поле задает бит адреса интерфейса HIF, используемый как бит 7 адреса колонки. Режим полуширины шины: это поле задает бит адреса интерфейса HIF, используемый как бит 8 адреса колонки. Режим четверти ширины шины: это поле задает бит адреса интерфейса HIF, используемый как бит 9 адреса колонки. Валидный диапазон: от 0 до 7, x и 31. x здесь указывает валидное значение во встроенной конфигурации ECC. Базовое внутреннее значение: 7. Выбранный бит адреса интерфейса HIF определяется путём прибавления базового внутреннего значения к значению этого поля. В режиме встроенной конфигурации ECC (MEMC_INLINE_ECC=1) и когда ECC подключен (ECCCFG0.ecc_mode>0), наибольшие три бита адресов колонок должны соотноситься с наибольшими из трёх валидных битов адреса интерфейса HIF. Если бит колонки 7 - третий	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		наибольший бит адреса колонки, он должен соотноситься с третьим наибольшим валидным битом адреса интерфейса HIF. (x = наибольший валидный бит адреса HIF - 2 - внутреннее базовое значение). Если это поле не используется, в него следует установить 31, и тогда в бит адреса колонки этого бита будет установлен 0. Режим программирования: статический.		
[15:13]	–	Зарезервировано.	0	
[20:16]	addrmap_col_b8	Режим полной ширины шины: это поле задает бит адреса интерфейса HIF, используемый как бит 8 адреса колонки. Режим полуширины шины: это поле задает бит адреса интерфейса HIF, используемый как бит 9 адреса колонки. Режим четверти ширины шины: это поле задает бит адреса интерфейса HIF, используемый как бит 11 адреса колонки (10 в режиме LPDDR2/LPDDR3). Валидный диапазон: от 0 до 7, x и 31. x здесь указывает валидное значение во встроенной конфигурации ECC. Базовое внутреннее значение: 8. Выбранный бит адреса интерфейса HIF определяется путём прибавления базового внутреннего значения к значению этого поля. Замечание: для спецификации JEDEC DDR2/3/mDDR, бит адреса колонки 10 зарезервирован для указания автопредзаряда, и таким образом, никакой бит адреса источника не может быть поставлен в соответствие биту адреса 10. В LPDDR2/LPDDR3, присутствует специально отведенный бит для автопредзаряда в шине CA и таким образом, бит 10 адреса колонки используется. В режиме встроенной конфигурации ECC (MEMC_INLINE_ECC=1) и когда ECC подключен (ECCCFG0.ecc_mode>0), биты наибольших трёх колонок должны соотноситься с наибольшими из трёх валидных битов адреса интерфейса HIF. Если бит колонки 8 - второй наибольший бит адреса колонки, он должен соотноситься со вторым наибольшим валидными битом адреса интерфейса HIF. (x = наибольший валидный бит адреса HIF - 1 - внутреннее базовое значение). Если бит колонки 8 - третий наибольший бит адреса колонки, он должен соотноситься с третьим наибольшим валидными битом адреса интерфейса HIF. (x = наибольший валидный бит адреса HIF - 2 - внутреннее базовое значение). Если это поле не используется, в него следует установить 31, и тогда в бит адреса колонки этого бита будет установлен 0. Режим программирования: статический.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
[23:21]	–	Зарезервировано.	0	
[28:24]	addrmap_col_b9	Режим полной ширины шины: это поле задает бит адреса интерфейса HIF, используемый как бит 9 адреса колонки. Режим полуширины шины: это поле задает бит адреса интерфейса HIF, используемый как бит 11 адреса колонки. (10 в режиме LPDDR2/LPDDR3). Режим четверти ширины шины: это поле задает бит адреса интерфейса HIF, используемый как бит 13 адреса колонки. (11 в режиме LPDDR2/LPDDR3). Валидный диапазон: от 0 до 7, x, и 31. x здесь указывает валидное значение во встроенной конфигурации ECC. Базовое внутреннее значение: 9. Выбранный бит адреса интерфейса HIF определяется путём прибавления базового внутреннего значения к значению этого поля. Замечание: для спецификации JEDEC DDR2/3/mDDR, бит 10 адреса колонки зарезервирован для указания автопредзаряда, и таким образом, никакой бит адреса источника не может быть поставлен в соответствие биту адреса 10. В LPDDR2/LPDDR3 присутствует специально отведенный бит для автопредзаряда в шине CA и таким образом, бит 10 адреса колонки используется. В режиме встроенной конфигурации ECC (MEMC_INLINE_ECC=1) и когда ECC подключен (ECCCFG0.ecc_mode>0), биты наибольших трёх колонок должны соотноситься с наибольшими из трёх валидных битов адреса интерфейса HIF. Если бит колонки 9 - наибольший бит адреса колонки, он должен соотноситься с наибольшим валидным битом адреса интерфейса HIF. (x = наибольший валидный бит адреса HIF - внутреннее базовое значение). Если бит колонки 9 - второй наибольший бит адреса колонки, он должен соотноситься со вторым наибольшим валидными битом адреса интерфейса HIF. (x = наибольший валидный бит адреса HIF - 1 - внутреннее базовое значение). Если бит колонки 9 - третий наибольший бит адреса колонки, он должен соотноситься с третьим наибольшим валидными битом адреса интерфейса HIF. (x = наибольший валидный бит адреса HIF - 2 - внутреннее базовое значение). Если это поле не используется, в него следует установить 31, и тогда в бит адреса колонки этого бита будет установлен 0. Режим программирования: статический.	0x0	R/W
[31:29]	–	Зарезервировано.	0	

24.6.86 Регистр соответствия адресов 4 (ADDRMAP4)

Поля регистра ADDRMAP4 описаны в Таблица 24.88.

Таблица 24.88. Регистр соответствия адресов 4

Диапазон	Название	Описание	Исходное значение	Тип доступа
[4:0]	addrmap_col_b10	Режим полной ширины шины: это поле задает бит адреса интерфейса HIF, используемый как бит 11 адреса колонки (10 в режиме LPDDR2/LPDDR3). Режим полуширины шины: это поле задает бит адреса интерфейса HIF, используемый как бит 13 адреса колонки (11 в режиме LPDDR2/LPDDR3). Режим четверти ширины шины: НЕ ИСПОЛЬЗУЕТСЯ. Позднее в этом описании будет дано значение, которое следует устанавливать, чтобы сделать его неиспользуемым. Валидный диапазон: от 0 до 7, x и 31. x здесь указывает валидное значение во встроенной конфигурации ECC. Базовое внутреннее значение: 10. Выбранный бит адреса интерфейса HIF определяется путём прибавления базового внутреннего значения к значению этого поля. Замечание: для спецификации JEDEC DDR2/3/mDDR, бит адреса колонки 10 зарезервирован для указания автопредзаряда, и таким образом, никакой бит адреса источника не может быть поставлен в соответствие биту адреса 10. В LPDDR2/LPDDR3 присутствует специально отведенный бит для автопредзаряда в шине CA и, таким образом, бит 10 адреса колонки используется. В режиме встроенной конфигурации ECC (MEMC_INLINE_ECC=1) и когда ECC подключен (ECCCFG0.ecc_mode>0), биты наибольших трёх колонок должны соотноситься с наибольшими из трёх валидных битов адреса интерфейса HIF. Если бит колонки 10 - наибольший бит адреса колонки, он должен соотноситься с наибольшим валидным битом адреса интерфейса HIF. (x = наибольший валидный бит адреса HIF - внутреннее базовое значение). Если бит колонки 10 - второй наибольший бит адреса колонки, он должен соотноситься со вторым наибольшим валидными битом адреса интерфейса HIF. (x = наибольший валидный бит адреса HIF - 1 - внутреннее базовое значение). Если бит колонки 10 - третий наибольший бит адреса колонки, он должен соотноситься с третьим наибольшим валидными битом адреса интерфейса HIF. (x = наибольший валидный бит адреса HIF - 2 - внутреннее базовое значение). Если это поле не используется, в него следует установить 31, и тогда в бит адреса колонки этого бита будет установлен 0.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Режим программирования: статический.		
[7:5]	–	Зарезервировано.	0	
[12:8]	addrmap_col_b11	Режим полной ширины шины: это поле задает бит адреса интерфейса HIF, используемый как бит 13 адреса колонки (11 в режиме LPDDR2/LPDDR3). Режим полуширины шины: НЕ ИСПОЛЬЗУЕТСЯ. Позднее в этом описании будет дано значение, которое следует устанавливать, чтобы сделать его неиспользуемым. Режим четверти ширины шины: НЕ ИСПОЛЬЗУЕТСЯ. Позднее в этом описании будет дано значение, которое следует устанавливать, чтобы сделать его неиспользуемым. Валидный диапазон: от 0 до 7, x и 31. x здесь указывает валидное значение во встроенной конфигурации ECC. Базовое внутреннее значение: 11. Выбранный бит адреса интерфейса HIF определяется путём прибавления базового внутреннего значения к значению этого поля. Замечание: для спецификации JEDEC DDR2/3/mDDR, бит адреса колонки 10 зарезервирован для указания автопредзаряда, и таким образом, никакой бит адреса источника не может быть поставлен в соответствие биту адреса 10. В LPDDR2/LPDDR3 присутствует специально отведенный бит для автопредзаряда в шине CA и, таким образом, бит 10 адреса колонки используется. В режиме встроенной конфигурации ECC (MEMC_INLINE_ECC=1) и когда ECC подключен (ECCCFG0.ecc_mode>0), биты наибольших трёх колонок должны соотноситься с наибольшими из трёх валидных битов адреса интерфейса HIF. Если бит колонки 11 - наибольший бит адреса колонки, он должен соотноситься с наибольшим валидными битом адреса интерфейса HIF. (x = наибольший валидный бит адреса HIF - внутреннее базовое значение). Если бит колонки 11 - второй наибольший бит адреса колонки, он должен соотноситься со вторым наибольшим валидными битом адреса интерфейса HIF. (x = наибольший валидный бит адреса HIF - 1 - внутреннее базовое значение). Если бит колонки 11 - третий наибольший бит адреса колонки, он должен соотноситься с третьим наибольшим валидными битом адреса интерфейса HIF. (x = наибольший валидный бит адреса HIF - 2 - внутреннее базовое значение). Если это поле не используется, в него следует установить 31, и тогда в бит адреса колонки этого бита будет установлен 0. Режим программирования: статический.	0x0	R/W
[31:13]	–	Зарезервировано.	0	

24.6.87 Регистр соответствия адресов 5 (ADDRMAP5)

Поля регистра ADDRMAP5 описаны в Таблица 24.89.

Таблица 24.89. Регистр соответствия адресов 5

Диапазон	Название	Описание	Исходное значение	Тип доступа
[3:0]	addrmap_row_b0	Задает биты адреса интерфейса HIF, используемые как бит 0 адреса строки. Валидный диапазон: от 0 до 11. Базовое внутреннее значение: 6. Выбранные биты адреса интерфейса HIF для каждого из битов адреса строки определяются путем прибавления внутреннего базового значения к значению этого поля. Режим программирования: статический.	0x0	R/W
[7:4]	–	Зарезервировано.	0	
[11:8]	addrmap_row_b1	Задает биты адреса интерфейса HIF, используемые как бит 1 адреса строки. Валидный диапазон: от 0 до 11. Базовое внутреннее значение: 7. Выбранные биты адреса интерфейса HIF для каждого из битов адреса строки определяются путем прибавления внутреннего базового значения к значению этого поля. Режим программирования: статический.	0x0	R/W
[15:12]	–	Зарезервировано.	0	
[19:16]	addrmap_row_b2_10	Задает биты адреса интерфейса HIF, используемые как биты адреса строки от 2 до 10. Валидный диапазон: от 0 до 11, и 15. Базовое внутреннее значение: 8 (для бита адреса строки 2), 9 (для бита адреса строки 3), 10 (для бита адреса строки 4) и так далее вплоть до 16-ти (для бита адреса строки 10). Выбранный бит адреса интерфейса HIF для каждого из битов адреса строки определяется путем прибавления внутреннего базового значения к значению этого поля. Когда в это поле установлено 15, значения битов адреса строки от 2 до 10-ти определяются регистрами ADDRMAP9, ADDRMAP10, ADDRMAP11. Режим программирования: статический.	0x0	R/W
[23:20]	–	Зарезервировано.	0	
[27:24]	addrmap_row_b11	Задает бит адреса интерфейса HIF, используемый как бит 11 адреса строки. Валидный диапазон: от 0 до 11, и 15. Базовое внутреннее значение: 17. Выбранный бит адреса интерфейса HIF определяется путём прибавления базового внутреннего значения к значению этого поля.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Если это поле не используется, следует установить в него 15, тогда в бит 11 адреса строки будет установлен 0. Режим программирования: статический.		
[31:28]	–	Зарезервировано.	0	

24.6.88 Регистр соответствия адресов 6 (ADDRMAP6)

Поля регистра ADDRMAP6 описаны в Таблица 24.90.

Таблица 24.90. Регистр соответствия адресов 6

Диапазон	Название	Описание	Исходное значение	Тип доступа
[3:0]	addrmap_row_b12	Задает бит адреса интерфейса HIF, используемый как бит 12 адреса строки. Валидный диапазон: от 0 до 11, и 15. Базовое внутреннее значение: 18. Выбранный бит адреса интерфейса HIF определяется путём прибавления базового внутреннего значения к значению этого поля. Если это поле не используется, следует установить в него 15 и тогда в бит 12 адреса строки будет установлен 0. Режим программирования: статический.	0x0	R/W
[7:4]	–	Зарезервировано.	0	
[11:8]	addrmap_row_b13	Задает бит адреса интерфейса HIF, используемый как бит 13 адреса строки. Валидный диапазон: от 0 до 11, и 15. Базовое внутреннее значение: 19. Выбранный бит адреса интерфейса HIF определяется путём прибавления базового внутреннего значения к значению этого поля. Если это поле не используется, следует установить в него 15 и тогда в бит 13 адреса строки будет установлен 0. Режим программирования: статический.	0x0	R/W
[15:12]	–	Зарезервировано.	0	
[19:16]	addrmap_row_b14	Задает бит адреса интерфейса HIF, используемый как бит 14 адреса строки. Валидный диапазон: от 0 до 11, и 15. Базовое внутреннее значение: 20. Выбранный бит адреса интерфейса HIF определяется путём прибавления базового внутреннего значения к значению этого поля. Если это поле не используется, следует установить в него 15 и тогда в бит 14 адреса строки будет установлен 0. Режим программирования: статический.	0x0	R/W
[23:20]	–	Зарезервировано.	0	
[27:24]	addrmap_row_b15	Задает бит адреса интерфейса HIF, используемый как бит 15 адреса строки. Валидный диапазон: от 0 до 11, и 15.	0x0	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Базовое внутреннее значение: 21. Выбранный бит адреса интерфейса HIF определяется путём прибавления базового внутреннего значения к значению этого поля. Если это поле не используется, следует установить в него 15 и тогда в бит 15 адреса строки будет установлен 0. Режим программирования: статический.		
[28]	–	Зарезервировано.	0	
[30:29]	lpddr4_6gb_12gb_24gb	Указывает, какой тип устройства LPDDR4 SDRAM используется. 2'b00: устройства LPDDR4 SDRAM 6/12/24 Гбит не используются. Все адреса валидны; 2'b01: используется устройство LPDDR4 SDRAM 6 Гбит. Каждый адрес, имеющий строку [14:13]==2'b11 считается невалидным; 2'b10: используется устройство LPDDR4 SDRAM 12 Гбит. Каждый адрес, имеющий строку [15:14]==2'b11 считается невалидным; 2'b11: используется устройство LPDDR4 SDRAM 24 Гбит. Каждый адрес, имеющий строку [16:15]==2'b11 считается невалидным; Присутствует только в реализациях, сконфигурированных для поддержки LPDDR4. Режим программирования: статический.	0x0	R/W
[31]	lpddr3_6gb_12gb	Следует устанавливать в это поле 1, если используется устройство LPDDR3 SDRAM на 6 или 12 Гбит. 1: используется устройство LPDDR3 SDRAM 6/12 Гбит. Каждый адрес, имеющий строку [14:13]==2'b11 считается невалидным; 0: используется устройство, не относящееся к LPDDR3 6/12 Гбит. Все адреса валидны. Присутствует только в реализациях, сконфигурированных для поддержки LPDDR3. Режим программирования: статический.	0x0	R/W

24.6.89 Регистр соответствия адресов 7 (ADDRMAP7)

Поля регистра ADDRMAP7 описаны в Таблица 24.91.

Таблица 24.91. Регистр соответствия адресов 7

Диапазон	Название	Описание	Исходное значение	Тип доступа
[3:0]	addrmap_row_b16	Это поле задает бит адреса интерфейса HIF, используемый как бит 16 адреса строки. Валидный диапазон: от 0 до 11, и 15. Базовое внутреннее значение: 22.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Выбранный бит адреса интерфейса HIF определяется путём прибавления базового внутреннего значения к значению этого поля. Если это поле не используется, в него следует установать 15, тогда в бит 16 адреса строки будет установлен 0. Режим программирования: статический.		
[7:4]	–	Зарезервировано.	0	
[11:8]	addrmap_row_b17	Это поле задает бит адреса интерфейса HIF, используемый как бит 17 адреса строки. Валидный диапазон: от 0 до 11, и 15. Базовое внутреннее значение: 23. Выбранный бит адреса интерфейса HIF определяется путём прибавления базового внутреннего значения к значению этого поля. Если это поле не используется, в него следует установать 15, тогда в бит 17 адреса строки будет установлен 0. Режим программирования: статический.	0x0	R/W
[31:12]	–	Зарезервировано.	0	

24.6.90 Регистр соответствия адресов 8 (ADDRMAP8)

Поля регистра ADDRMAP8 описаны в Таблица 24.92.

Таблица 24.92. Регистр соответствия адресов 8

Диапазон	Название	Описание	Исходное значение	Тип доступа
[5:0]	addrmap_bg_b0	Это поле задает биты адреса интерфейса HIF, используемые как бит 0 адреса группы банков. Валидный диапазон: от 0 до 32, и 63. Базовое внутреннее значение: 2. Выбранный бит адреса интерфейса HIF для каждого бита адреса группы банков определяется путём прибавления базового внутреннего значения к значению этого поля. Если это поле не используется, в него следует установать 63, тогда в бит 0 адреса группы банков будет установлен 0. Режим программирования: статический.	0x0	R/W
[7:6]	–	Зарезервировано.	0	
[13:8]	addrmap_bg_b1	Это поле задает биты адреса интерфейса HIF, используемые как бит 1 адреса группы банков. Валидный диапазон: от 0 до 32, и 63. Базовое внутреннее значение: 3. Выбранный бит адреса интерфейса HIF для каждого бита адреса группы банков определяется путём прибавления базового внутреннего значения к значению этого поля. Если это поле не используется, в него следует установать 63, тогда в бит 1 адреса группы банков будет установлен 0.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Режим программирования: статический.		
[31:14]	–	Зарезервировано.	0	

24.6.91 Регистр соответствия адресов 9 (ADDRMAP9)

Поля регистра ADDRMAP9 описаны в Таблица 24.93.

Таблица 24.93. Регистр соответствия адресов 9

Диапазон	Название	Описание	Исходное значение	Тип доступа
[3:0]	addrmap_row_b2	Это поле задает биты адреса интерфейса HIF, используемые как бит 2 адреса строки. Валидный диапазон: от 0 до 11. Базовое внутреннее значение: 8. Выбранный бит адреса интерфейса HIF для каждого бита адреса строки определяется путём прибавления базового внутреннего значения к значению этого поля. Это поле регистра используется только когда в ADDRMAP5.addrmap_row_b2_10 установлено значение 15. Режим программирования: статический.	0x0	R/W
[7:4]	–	Зарезервировано.	0	
[11:8]	addrmap_row_b3	Это поле задает биты адреса интерфейса HIF, используемые как бит 3 адреса строки. Валидный диапазон: от 0 до 11. Базовое внутреннее значение: 9. Выбранный бит адреса интерфейса HIF для каждого бита адреса строки определяется путём прибавления базового внутреннего значения к значению этого поля. Это поле регистра используется только когда в ADDRMAP5.addrmap_row_b2_10 установлено значение 15. Режим программирования: статический.	0x0	R/W
[15:12]	–	Зарезервировано.	0	
[19:16]	addrmap_row_b4	Это поле задает биты адреса интерфейса HIF, используемые как бит 4 адреса строки. Валидный диапазон: от 0 до 11. Базовое внутреннее значение: 10. Выбранный бит адреса интерфейса HIF для каждого бита адреса строки определяется путём прибавления базового внутреннего значения к значению этого поля. Это поле регистра используется только когда в ADDRMAP5.addrmap_row_b2_10 установлено значение 15. Режим программирования: статический.	0x0	R/W
[23:20]	–	Зарезервировано.	0	
[27:24]	addrmap_row_b5	Это поле задает биты адреса интерфейса HIF, используемые как бит 5 адреса строки. Валидный диапазон: от 0 до 11. Базовое внутреннее значение: 11.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Выбранный бит адреса интерфейса HIF для каждого бита адреса строки определяется путём прибавления базового внутреннего значения к значению этого поля. Это поле регистра используется только когда в ADDRMAP5.addrmap_row_b2_10 установлено значение 15. Режим программирования: статический.		
[31:28]	–	Зарезервировано.	0	

24.6.92 Регистр соответствия адресов 10 (ADDRMAP10)

Поля регистра ADDRMAP10 описаны в Таблица 24.94.

Таблица 24.94. Регистр соответствия адресов 10

Диапазон	Название	Описание	Исходное значение	Тип доступа
[3:0]	addrmap_row_b6	Это поле задает биты адреса интерфейса HIF, используемые как бит 6 адреса строки. Валидный диапазон: от 0 до 11. Базовое внутреннее значение: 12. Выбранный бит адреса интерфейса HIF для каждого бита адреса строки определяется путём прибавления базового внутреннего значения к значению этого поля. Это поле регистра используется только когда в ADDRMAP5.addrmap_row_b2_10 установлено значение 15. Режим программирования: статический.	0x0	R/W
[7:4]	–	Зарезервировано.	0	
[11:8]	addrmap_row_b7	Это поле задает биты адреса интерфейса HIF, используемые как бит 7 адреса строки. Валидный диапазон: от 0 до 11. Базовое внутреннее значение: 13. Выбранный бит адреса интерфейса HIF для каждого бита адреса строки определяется путём прибавления базового внутреннего значения к значению этого поля. Это поле регистра используется только когда в ADDRMAP5.addrmap_row_b2_10 установлено значение 15. Режим программирования: статический.	0x0	R/W
[15:12]	–	Зарезервировано.	0	
[19:16]	addrmap_row_b8	Это поле задает биты адреса интерфейса HIF, используемые как бит 8 адреса строки. Валидный диапазон: от 0 до 11. Базовое внутреннее значение: 14. Выбранный бит адреса интерфейса HIF для каждого бита адреса строки определяется путём прибавления базового внутреннего значения к значению этого поля.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Это поле регистра используется только когда в ADDRMAP5.addrmap_row_b2_10 установлено значение 15. Режим программирования: статический.		
[23:20]	–	Зарезервировано.	0	
[27:24]	addrmap_row_b9	Это поле задает биты адреса интерфейса HIF, используемые как бит 9 адреса строки. Валидный диапазон: от 0 до 11. Базовое внутреннее значение: 15. Выбранный бит адреса интерфейса HIF для каждого бита адреса строки определяется путём прибавления базового внутреннего значения к значению этого поля. Это поле регистра используется только когда в ADDRMAP5.addrmap_row_b2_10 установлено значение 15. Режим программирования: статический.	0x0	R/W
[31:28]	–	Зарезервировано.	0	

24.6.93 Регистр соответствия адресов 11 (ADDRMAP11)

Поля регистра ADDRMAP11 описаны в Таблица 24.95.

Таблица 24.95. Регистр соответствия адресов 11

Диапазон	Название	Описание	Исходное значение	Тип доступа
[3:0]	addrmap_row_b10	Это поле задает биты адреса интерфейса HIF, используемые как бит 10 адреса строки. Валидный диапазон: от 0 до 11. Базовое внутреннее значение: 16. Выбранный бит адреса интерфейса HIF для каждого бита адреса строки определяется путём прибавления базового внутреннего значения к значению этого поля. Это поле используется только когда в ADDRMAP5.addrmap_row_b2_10 установлено значение 15. Режим программирования: статический.	0x0	R/W
[31:4]	–	Зарезервировано.	0	

24.6.94 Регистр конфигурирования ODT (ODTCFG)

Поля регистра ODTCFG описаны в Таблица 24.96.

Таблица 24.96. Регистр конфигурирования ODT

Диапазон	Название	Описание	Исходное значение	Тип доступа
[1:0]	–	Зарезервировано.	0	
[6:2]	rd_odt_delay	Задержка в тактах синхросигнала PHY в DFI, от момента отправки команды чтения до момента	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		установки значений терминации ODT, соответствующих этой команде. Установки ODT должны оставаться неизменными в течение всего времени, пока сигнал DQS активируется через uMCTL2. Рекомендованные значения: DDR2: CL + AL - 4 (не DDR2-1066), CL + AL - 5 (DDR2-1066). Если (CL + AL - 4 < 0), uMCTL2 не поддерживает терминацию ODT для операций чтения. DDR3: CL - CWL. DDR4: CL - CWL - RD_PREAMBLE + WR_PREAMBLE + DFITMG1.dfi_t_cmd_lat (для подстройки под режим CAL). WR_PREAMBLE = 1 (начальная часть записи 1tCK), 2 (начальная часть записи 2tCK). RD_PREAMBLE = 1 (начальная часть чтения 1tCK), 2 (начальная часть чтения 2tCK). Если (CL - CWL - RD_PREAMBLE + WR_PREAMBLE) < 0, uMCTL2 не поддерживает терминацию ODT для операций чтения. LPDDR3: RL + RD(tDQSCK(min)/tCK) - 1 - RU(tODTon(max)/tCK). Единицы измерения: такты синхросигнала PHY в DFI. Режим программирования: квазидинамические группы 1 и 4.		
[7]	–	Зарезервировано.	0	
[11:8]	rd_odt_hold	Количество тактов синхросигнала PHY в DFI для удержания терминации ODT для команды чтения. Минимальное поддерживаемое значение: 2. Рекомендованные значения: DDR2: BL8: 0x6 (не DDR2-1066), 0x7 (DDR2-1066). BL4: 0x4 (не DDR2-1066), 0x5 (DDR2-1066). DDR3: BL8 - 0x6 DDR4: BL8: 5 + RD_PREAMBLE. RD_PREAMBLE = 1 (начальная часть чтения 1tCK), 2 (начальная часть чтения 2tCK). LPDDR3:	0x4	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		BL8: 5 + RU(tDQSCK(max)/tCK) - RD(tDQSCK(min)/tCK) + RU(tODTon(max)/tCK). Единицы измерения: такты синхросигнала PHY в DFI. Режим программирования: квазидинамические группы 1 и 4.		
[15:12]	–	Зарезервировано.	0	
[20:16]	wr_odt_delay	Задержка в тактах синхросигнала PHY в DFI от момента отправки команды записи до момента установки значений терминации ODT, соответствующих этой команде. Установки ODT должны оставаться неизменными в течение всего времени, пока сигнал DQS активируется через uMCTL2. Рекомендованные значения: DDR2: CWL + AL - 3 (DDR2-400/533/667), CWL + AL - 4(DDR2-800), CWL + AL - 5 (DDR2-1066). Если (CWL + AL - 3 < 0), uMCTL2 не поддерживает терминацию ODT для операций записи. DDR3: 0x0. DDR4: DFITMG1.dfi_t_cmd_lat (для подстройки под режим CAL). LPDDR3: WL - 1 - RU(tODTon(max)/tCK)). Единицы измерения: такты синхросигнала PHY в DFI. Режим программирования: квазидинамические группы 1 и 4.	0x0	R/W
[23:21]	–	Зарезервировано.	0	
[27:24]	wr_odt_hold	Количество тактов синхросигнала PHY в DFI, необходимых для удержания терминации ODT для команды записи. Минимальное поддерживаемое значение: 2. Рекомендованные значения: DDR2: BL8: 0x5 (DDR2-400/533/667), 0x6 (DDR2-800), 0x7 (DDR2-1066). BL4: 0x3 (DDR2-400/533/667), 0x4 (DDR2-800), 0x5 (DDR2-1066). DDR3: BL8: 0x6. DDR4: BL8: 5 + WR_PREAMBLE + CRC_MODE.	0x4	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		WR_PREAMBLE = 1 (начальная часть записи 1tCK), 2 (начальная часть записи 2tCK). CRC_MODE = 0 (вне режима CRC), 1 (в режиме CRC). LPDDR3: BL8: 7 + RU(tODTon(max)/tCK) Единицы измерения: такты синхросигнала PHY в DFI. Режим программирования: квазидинамические группы 1 и 4.		
[31:28]	–	Зарезервировано.	0	

24.6.95 Регистр соответствия терминации ODT/ранга (ODTMAP)

Поля регистра ODTMAP описаны в Таблица 24.97.

Таблица 24.97. Регистр соответствия терминации ODT/ранга

Диапазон	Название	Описание	Исходное значение	Тип доступа
[1:0]	rank0_wr_odt	Указывает, какие удаленные терминации ODT обязаны быть включены в процессе записи в ранг 0. Каждый ранг имеет удаленную терминацию ODT (в SDRAM) которая может быть включена путём установки в этом регистре соответствующего бита. Ранг 0 контролируется младшим битом (LSB), ранг 1 контролируется битом, следующим после младшего и так далее. Для каждого ранга следует установить 1 в его бит для подключения его терминации ODT. Режим программирования: статический.	0x1	R/W
[3:2]	–	Зарезервировано.	0	
[5:4]	rank0_rd_odt	Указывает, какие удаленные терминации ODT обязаны быть включены в процессе чтения из ранга 0. Каждый ранг имеет удаленную терминацию ODT (в SDRAM) которая может быть включена путём установки в этом регистре соответствующего бита. Ранг 0 контролируется младшим битом (LSB), ранг 1 контролируется битом, следующим после младшего и так далее. Для каждого ранга следует установить 1 в его бит для подключения его терминации ODT. Режим программирования: статический.	0x1	R/W
[7:6]	–	Зарезервировано.	0	
[9:8]	rank1_wr_odt	Указывает, какие удаленные терминации ODT обязаны быть включены в процессе записи в ранг 1. Каждый ранг имеет удаленную терминацию ODT (в SDRAM) которая может быть включена путём установки в этом регистре соответствующего бита. Ранг 0 контролируется младшим битом (LSB), ранг 1 контролируется битом, следующим после младшего и так далее.	0x2	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Для каждого ранга следует установить 1 в его бит для подключения его терминации ODT. Присутствует только в конфигурациях, имеющих 2 и более рангов. . Режим программирования: статический.		
[11:10]	–	Зарезервировано.	0	
[13:12]	rank1_rd_odt	Указывает, какие удаленные терминации ODT обязаны быть включены в процессе чтения из ранга 1. Каждый ранг имеет удаленную терминацию ODT (в SDRAM) которая может быть включена путём установки в этом регистре соответствующего бита. Ранг 0 контролируется младшим битом (LSB), ранг 1 контролируется битом, следующим после младшего и так далее. Для каждого ранга следует установить 1 в его бит для подключения его терминации ODT. Присутствует только в конфигурациях, имеющих 2 или более рангов. Режим программирования: статический.	0x2	R/W
[31:14]	–	Зарезервировано.	0	

24.6.96 Регистр управления планировщиком (SCHED)

Поля регистра SCHED описаны в Таблица 24.98.

Таблица 24.98. Регистр управления планировщиком

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	force_low_pri_n	Активный сигнал низкого уровня. Когда этот бит установлен ('0'), всем входящим транзакциям присваивается низкий приоритет. Это подразумевает, что все команды высокоприоритетных чтений (High Priority Read (HPR)) и чтений варьируемого приоритета (Variable Priority Read (VPR)) будут обрабатываться как команды низкоприоритетных чтений (Low Priority Read (LPR)). Что касается записей, все команды варьируемого приоритета записи (Variable Priority Write (VPW)) будут обрабатываться как команды нормального приоритета записи (Normal Priority Write (NPW)). Приведение входящих транзакций к низкому приоритету также автоматически отключает путь обхода (Bypass path) для команд чтения. ИСПОЛЬЗОВАТЬ ТОЛЬКО В РАБОЧЕМ РЕЖИМЕ. Режим программирования: статический.	0x1	R/W
[1]	prefer_write	Если этот бит установлен, селектор банков ставит записи в приоритет над чтениями. ИСПОЛЬЗОВАТЬ ТОЛЬКО В РЕЖИМЕ ОТЛАДКИ.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Режим программирования: статический.		
[2]	pageclose	Если этот бит установлен, банк удерживается открытым только когда в CAM к этому банку присутствуют транзакции обращения к странице. Последняя команда записи или чтения в CAM с банком и обращением к странице будет исполнена с автопредзарядом если SCHED1.pageclose_timer=0. Даже если в этот регистр установлена 1, а в SCHED1.pageclose_timer установлен 0, явно заданный предзаряд (а не автопредзаряд) может быть отправлен в некоторых случаях, когда присутствует переключение режима между записью или чтением или между низкоприоритетным и высокоприоритетным чтением. Команды чтения и записи, которые исполняются как часть запросов очистки (scrub requests) ECC также исполняются без автопредзаряда. Если значение этого бита 0, банк остается открытым до тех пор, пока не возникнет необходимость закрыть его (чтобы открыть другую страницу, или для таймаута страницы или таймаута обновления) - это также известно как стратегия открытой страницы. Стратегия открытой страницы (open page policy) может быть замещена путем установки бита покомандного автопредзаряда (per-command-autopre bit) в интерфейсе HIF (hif_cmd_autopre). Функция закрытия страницы предоставляет компромисс между стратегиями открытой и закрытой страницы. ИСПОЛЬЗОВАТЬ ТОЛЬКО В РАБОЧЕМ РЕЖИМЕ. Режим программирования: квазидинамическая группа 3.	0x1	R/W
[7:3]	–	Зарезервировано.	0	
[12:8]	lpr_num_entries	Значение этого бита плюс один отражает число записей в хранилище низкоприоритетных транзакций (MEMC_NO_OF_ENTRY - (SCHED.lpr_num_entries + 1)) - число записей, доступных для хранилища высокоприоритетных транзакций. Установка в это поле максимального значения определяет все записи в хранилище низкоприоритетных транзакций. Установка в это поле 0 определяет одну запись в хранилище низкоприоритетных транзакций, а остальные - в хранилище высокоприоритетных транзакций. Замечание: в конфигурациях ECC количество отправляемых разрешений на передачу записей и низкоприоритетных чтений - на одну меньше, чем в конфигурациях без ECC. Каждая единичная запись резервируется в CAMs для записей и низкоприоритетных чтений для хранения запросов	0x10	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		RMW, возникающих при операции однобитной коррекции RMW. Режим программирования: статический.		
[15:13]	–	Зарезервировано.	0	
[23:16]	go2critical_hysteresis	НЕ ИСПОЛЬЗУЕТСЯ. Режим программирования: статический.	0x0	R/W
[30:24]	rdwr_idle_gap	Когда приоритетное хранилище транзакций остается пустым на протяжении этого числа тактов синхросигнала, следует переключиться на альтернативное хранилище транзакций, если оно не является пустым. Хранилище транзакций чтений (как высокого, так и низкого приоритетов) является приоритетным хранилищем транзакций по умолчанию, а хранилище транзакций записей - альтернативным. Когда установлен приоритет записей над чтениями, этот порядок инвертируется. 0x0 - разрешенное значение для этого поля. Когда в него установлен 0x0, то, как только условия переключения хранилищ транзакций станут истинны, немедленно будет произведено переключение. ИСПОЛЬЗОВАТЬ ТОЛЬКО В РАБОЧЕМ РЕЖИМЕ. Единицы измерения: такты синхросигнала DFI. Режим программирования: статический.	0x0	R/W
[31]	–	Зарезервировано.	0	

24.6.97 Регистр управления планировщиком 1 (SCHED1)

Поля регистра SCHED1 описаны в Таблица 24.99.

Таблица 24.99. Регистр управления планировщиком 1

Диапазон	Название	Описание	Исходное значение	Тип доступа
[7:0]	pageclose_timer	Это поле работает в связке с полем закрытия страницы регистра SCHED: SCHED.pageclose. Оно имеет смысл только если SCHED.pageclose==1. Если SCHED.pageclose==1 и pageclose_timer==0, тогда для последней команды записи или чтения в CAM с обращением к банку и странице (bank and page hit) может быть запланирован автопредзаряд. Следует отметить, что иногда вместо авто-предзаряда планируется явно заданный предзаряд. Подробнее условия, при которых это может произойти, можно узнать в описании поля SCHED.pageclose. Если SCHED.pageclose==1 и pageclose_timer>0, тогда для последней команды записи или чтения в CAM с	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		обращением к банку и странице (bank and page hit) авто-предзаряд не планируется. Вместо этого запускается таймер с внутренним значением, равным pageclose_timer. Существует таймер на по-банковой основе. Этот таймер декрементируется, если только следующая операция записи или чтения в CAM к банку не является обращением к странице. Если следующая операция записи или чтения в CAM к банку является обращением к странице, этот таймер сбрасывается к значению pageclose_timer. Когда таймер достигает 0, производится попытка запланировать явно заданный предзаряд. Единицы измерения: такты синхросигнала DFI. Режим программирования: статический.		
[31:8]	–	Зарезервировано.	0	

24.6.98 Регистр CAM чтений с высоким приоритетом 1 (PERFHPR1)

Поля регистра PERFHPR1 описаны в Таблица 24.100.

Таблица 24.100. Регистр CAM чтений с высоким приоритетом 1

Диапазон	Название	Описание	Исходное значение	Тип доступа
[15:0]	hpr_max_starve	Количество тактов DFI, в течение которых очередь HPR может находиться в состоянии зависания до того, как зависание станет критическим. Минимальное валидное функциональное значение для этого поля - 0x1. Программирование в это поле 0x0 отключит функционал работы с зависаниями; в процессе нормальной работы, эту функцию не следует отключать, поскольку это приведёт к избыточным запаздываниям. ИСПОЛЬЗОВАТЬ ТОЛЬКО В РАБОЧЕМ РЕЖИМЕ. Единицы измерения: такты синхросигнала DFI. Режим программирования: квазидинамическая группа 3.	0x1	R/W
[23:16]	–	Зарезервировано.	0	
[31:24]	hpr_xact_run_length	Число транзакций, которые обрабатываются с того момента, как очередь HPR переходит в состояние критического зависания, является наименьшей величиной из: - a) значения этого поля; - b) числа доступных транзакций. Единицы измерения: транзакции. ИСПОЛЬЗОВАТЬ ТОЛЬКО В РАБОЧЕМ РЕЖИМЕ. Режим программирования: квазидинамическая группа 3.	0xf	R/W

24.6.99 Регистр CAM чтений с низким приоритетом 1 (PERFLPR1)

Поля регистра PERFLPR1 описаны в Таблица 24.101.

Таблица 24.101. Регистр CAM чтений с низким приоритетом 1

Диапазон	Название	Описание	Исходное значение	Тип доступа
[15:0]	lpr_max_starve	Количество тактов DFI, в течение которых очередь LPR (чтений с низким приоритетом) может находиться в состоянии зависания до того, как зависание станет критическим. Минимальное валидное функциональное значение для этого поля - 0x1. Программирование в это поле 0x0 отключит функционал работы с зависаниями; в процессе нормальной работы эту функцию не следует отключать, поскольку это приведёт к избыточным запаздываниям. ИСПОЛЬЗОВАТЬ ТОЛЬКО В РАБОЧЕМ РЕЖИМЕ. Единицы измерения: такты синхросигнала DFI. Режим программирования: квазидинамическая группа 3.	0x7f	R/W
[23:16]	–	Зарезервировано.	0	
[31:24]	lpr_xact_run_length	Число транзакций, которые обрабатываются с того момента, как очередь LPR переходит в состояние критического зависания, является наименьшей величиной из: - a) значения этого поля; - b) числа доступных транзакций. Единицы измерения: транзакции. ИСПОЛЬЗОВАТЬ ТОЛЬКО В РАБОЧЕМ РЕЖИМЕ. Режим программирования: квазидинамическая группа 3.	0xf	R/W

24.6.100 Регистр записей CAM 1 (PERFWR1)

Поля регистра PERFWR1 описаны в Таблица 24.102.

Таблица 24.102. Регистр записей CAM 1

Диапазон	Название	Описание	Исходное значение	Тип доступа
[15:0]	w_max_starve	Количество тактов синхросигнала DFI, в течение которого очередь WR (записи) может находиться в состоянии зависания перед тем, как она перейдёт в критическое состояние. Минимальное валидное функциональное значение для этого поля - 0x1. Программирование в него 0x0 отключит функционал обработки зависаний; в процессе нормальной работы эта функция не должна быть отключена, поскольку это может спровоцировать избыточные запаздывания. ИСПОЛЬЗОВАТЬ ТОЛЬКО В РАБОЧЕМ РЕЖИМЕ.	0x7f	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Единицы измерения: такты синхросигнала DFI. Режим программирования: квазидинамическая группа 3.		
[23:16]	–	Зарезервировано.	0	
[31:24]	w_xact_run_length	Количество транзакций, которые обрабатываются с момента, когда очередь WR переходит в критическое состояние - это наименьшее из: а) значения этого поля; б) числа доступных транзакций. Единицы измерения: транзакции. ИСПОЛЬЗОВАТЬ ТОЛЬКО В РАБОЧЕМ РЕЖИМЕ. Режим программирования: квазидинамическая группа 3.	0xf	R/W

24.6.101 Регистр отладки 0 (DBG0)

Поля регистра DBG0 описаны в Таблица 24.103.

Таблица 24.103. Регистр отладки 0

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	dis_wc	Когда это поле имеет значение 1, оно отключает объединение записи (write combine). ИСПОЛЬЗОВАТЬ ТОЛЬКО В РЕЖИМЕ ОТЛАДКИ. Режим программирования: статический.	0x0	R/W
[3:1]	–	Зарезервировано.	0	
[4]	dis_collision_page_opt	Когда в это поле установлен 0, то для сброшенных команд, в случае коллизий, отключается функция автопредзаряда. Случаи коллизий - это записи, за которыми следуют чтения к одному и тому же адресу; чтения, за которыми следуют записи к одному и тому же адресу; или записи, за которыми следуют записи к одному и тому же адресу при условии, что бит DBG0.dis_wc bit = 1 (случаи, когда сравнения одного и того же адреса исключают два бита адреса, отражающих критическое слово). ИСПОЛЬЗОВАТЬ ТОЛЬКО В РЕЖИМЕ ОТЛАДКИ. Режим программирования: статический.	0x0	R/W
[31:5]	–	Зарезервировано.	0	

24.6.102 Регистр отладки 1 (DBG1)

Поля регистра DBG1 описаны в Таблица 24.104.

Таблица 24.104. Регистр отладки 1

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	dis_dq	Когда в это поле установлена 1, uMCTL2 не будет исключать из очереди никакие транзакции от CAM. Обход также отключен. Все транзакции собираются в очередь в CAM. Пока это поле установлено, к SDRAM не производится отправка никаких записей или чтений. Этот бит может использоваться для предотвращения отправки чтений или записей от uMCTL2, что делает безопасной модификацию полей определенных регистров, относящихся к чтениям и записям (подробнее об этом см. Руководство Пользователя). После установки этого бита, настойчиво рекомендуется произвести опрос полей DBGCAM.wr_data_pipeline_empty и DBGCAM.rd_data_pipeline_empty, перед изменением любых регистров, влияющих на чтения и записи. Это позволит удостовериться, что соответствующая им логика в DDRC находится в состоянии бездействия. Этот бит спроектирован так, чтобы переключать его в процессе работы (on-the-fly). Режим программирования: динамический.	0x0	R/W
[1]	dis_hif	Когда это поле имеет значение 1, uMCTL2 устанавливает сигнал команды интерфейса HIF hif_cmd_stall. uMCTL2 будет игнорировать hif_cmd_valid и все прочие связанные с этим сигналы запросов. Этот бит спроектирован так, чтобы переключать его в процессе работы (on-the-fly). Режим программирования: динамический.	0x0	R/W
[31:2]	–	Зарезервировано.	0	

24.6.103 Регистр отладки CAM (DBGCAM)

Поля регистра DBGCAM описаны в Таблица 24.105.

Таблица 24.105. Регистр отладки CAM

Диапазон	Название	Описание	Исходное значение	Тип доступа
[5:0]	dbg_hpr_q_depth	Глубина очереди высокоприоритетных чтений. ИСПОЛЬЗОВАТЬ ТОЛЬКО В РЕЖИМЕ ОТЛАДКИ. Режим программирования: динамический.	0x0	R
[7:6]	–	Зарезервировано.	0	
[13:8]	dbg_lpr_q_depth	Глубина очереди низкоприоритетных чтений. Последняя запись очереди LPR зарезервирована для операции ECC SCRUB (error-correcting code, корректирующий код). Эта запись не включается в расчёт глубины очереди. ИСПОЛЬЗОВАТЬ ТОЛЬКО В РЕЖИМЕ ОТЛАДКИ.	0x0	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Режим программирования: динамический.		
[15:14]	–	Зарезервировано.	0	
[21:16]	dbg_w_q_depth	Глубина очереди записей. Последняя запись очереди WR зарезервирована для операции ECC SCRUB. Эта запись не включается в расчёт глубины очереди. ИСПОЛЬЗОВАТЬ ТОЛЬКО В РЕЖИМЕ ОТЛАДКИ. Режим программирования: динамический.	0x0	R
[23:22]	–	Зарезервировано.	0	
[24]	dbg_stall	Режим останова (Stall). ИСПОЛЬЗОВАТЬ ТОЛЬКО В РЕЖИМЕ ОТЛАДКИ. Режим программирования: динамический.	0x0	R
[25]	dbg_rd_q_empty	Когда этот бит имеет значение 1, все очереди команд чтения и буферы данных чтения в DDRC пусты. Это поле должно использоваться только для отладки. Пример сценария использования: когда контроллер входит в режим саморегенерации с использованием алгоритма входа при низком питании (Low-Power entry sequence), ожидается, что контроллер исполнил все команды в очередях и данные записи и чтения опустошены. Следовательно, в такой момент это поле должно иметь значение 1. ИСПОЛЬЗОВАТЬ ТОЛЬКО В РЕЖИМЕ ОТЛАДКИ. Режим программирования: динамический.	0x0	R
[26]	dbg_wr_q_empty	Когда этот бит имеет значение 1, все очереди команд записи и буферы данных записи в DDRC пусты. Это поле должно использоваться только для отладки. Пример сценария использования: когда контроллер входит в режим саморегенерации с использованием алгоритма входа при низком питании, ожидается, что контроллер исполнил все команды в очередях и данные записи и чтения опустошены. Следовательно, в такой момент это поле должно иметь значение 1. ИСПОЛЬЗОВАТЬ ТОЛЬКО В РЕЖИМЕ ОТЛАДКИ. Режим программирования: динамический.	0x0	R
[27]	–	Зарезервировано.	0	
[28]	rd_data_pipeline_empty	Этот бит указывает, что конвейер данных чтения в интерфейсе DFI пуст. Предполагается, что к этому полю производится по меньше мере два опроса (poll) после установки DBG1.dis_dq, для того, чтобы удостовериться, что все оставшиеся команды/данные завершены. Режим программирования: динамический.	0x0	R
[29]	wr_data_pipeline_empty	Этот бит указывает, что конвейер данных записи в интерфейсе DFI пуст. Предполагается, что к этому полю производится по меньше мере два опроса (poll) после установки DBG1.dis_dq, для того, чтобы удостовериться, что все оставшиеся команды/данные завершены.	0x0	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Режим программирования: динамический.		
[31:30]	–	Зарезервировано.	0	

24.6.104 Регистр отладки команд (DBGCMD)

Поля регистра DBGCMD описаны в Таблица 24.106.

Таблица 24.106. Регистр отладки команд

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	rank0_refresh	Установка 1 в этот бит регистра указывает uMCTL2 отправить команду регенерации к рангу 0. Запись в этот бит провоцирует установку DBGSTAT.rank0_refresh_busy. Когда DBGSTAT.rank0_refresh_busy очищается, это значит, что команда сохранена в uMCTL2. В конфигурации 3DS, команда регенерации отправляется к рангу индекса 0. Эта операция может быть произведена только когда RFSHCTL3.dis_auto_refresh=1. Рекомендуется НЕ устанавливать этот бит, если система находится в режиме инициализации, глубоком режиме низкого питания или режиме максимального энергосбережения. Контролепригодность: readOnly. Режим программирования: динамический.	0x0	R/W
[1]	rank1_refresh	Установка 1 в этот бит регистра указывает uMCTL2 отправить команду регенерации к рангу 1.Запись в этот бит провоцирует установку DBGSTAT.rank1_refresh_busy. Когда DBGSTAT.rank1_refresh_busy очищается, это значит, что команда сохранена в uMCTL2. В конфигурации 3DS, команда регенерации отправляется к рангу индекса 1. Эта операция может быть произведена только когда RFSHCTL3.dis_auto_refresh=1. Рекомендуется НЕ устанавливать этот бит, если система находится в режиме инициализации, глубоком режиме низкого питания или режиме максимального энергосбережения. Контролепригодность: readOnly. Режим программирования: динамический.	0x0	R/W
[3:2]	rank2_refresh	Зарезервировано.	0	
[4]	zq_calib_short	Установка 1 в этот бит регистра указывает uMCTL2 отправить команду ZQCS (короткая калбировка ZQ)/MPC(калибровка ZQ) к SDRAM. Когда это запрос сохраняется в uMCTL2, этот бит автоматически очищается. Эта операция может быть произведена только когда ZQCTL0.dis_auto_zq=1. Рекомендуется НЕ устанавливать этот бит, если система находится в	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		режиме инициализации, саморегенерации (кроме LPDDR4), саморегенерации при низком питании (LPDDR4), глубоком режиме низкого питания или режиме максимального энергосбережения. При саморегенерации (кроме LPDDR4) или саморегенерации при низком питании (LPDDR4) такая команда будет запланирована после выхода из этих состояний. В глубоком режиме низкого питания и режиме максимального энергосбережения, эта команда не будет запланирована, хотя DBGSTAT.zq_calib_short_busy будет снят. Контролепригодность: readOnly. Режим программирования: динамический.		
[5]	ctrlupd	Установка 1 в этот бит регистра указывает uMCTL2 отправить dfi_ctrlupd_req к PHY. Когда этот запрос загружается в uMCTL2, этот бит очищается автоматически. Эта операция должна производиться только когда DFIUPD0.dis_auto_ctrlupd=1. Контролепригодность: readOnly. Режим программирования: динамический.	0x0	R/W
[31:6]	–	Зарезервировано.	0	

24.6.105 Регистр статуса отладки (DBGSTAT)

Поля регистра DBGSTAT описаны в Таблица 24.107.

Таблица 24.107. Регистр статуса отладки

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	rank0_refresh_busy	Система SoC может инициировать операцию rank0_refresh (операцию регенерации к рангу 0) только если этот сигнал имеет низкое значение. Этот сигнал получает высокое значение спустя такт после того, как в DBGCMD.rank0_refresh установлена "1". Он получает низкое значение, когда операция rank0_refresh сохранена в uMCTL2. Рекомендуется не производить операций rank0_refresh, когда этот сигнал имеет высокое значение. 0 - указывает, что система SoC может инициировать операцию rank0_refresh; 1 - указывает, что операция rank0_refresh еще не сохранена в uMCTL2. Режим программирования: динамический.	0x0	R
[1]	rank1_refresh_busy	Система SoC может инициировать операцию rank1_refresh (операции регенерации к рангу 1) только если этот сигнал имеет низкое значение. Этот	0x0	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
		сигнал получает высокое значение спустя такт после того, как в DBGCMD.rank1_refresh установлена "1". Он получает низкое значение, когда операция rank1_refresh сохранена в uMCTL2. Рекомендуется не производить операций rank1_refresh, когда этот сигнал имеет высокое значение. 0 - указывает, что система SoC может инициировать операцию rank1_refresh; 1 - указывает, что операция rank1_refresh еще не сохранена в uMCTL2. Режим программирования: динамический.		
[3:2]	–	Зарезервировано.	0	
[4]	zq_calib_short_busy	Система SoC может инициировать операцию ZQCS (короткая калибровка ZQ) только если этот сигнал имеет низкое значение. Этот сигнал получает высокое значение спустя такт после того, как uMCTL2 принимает запрос на ZQCS. Он получает низкое значение, когда операция ZQCS инициирована в uMCTL2. Рекомендуется не производить операций ZQCS, когда этот сигнал имеет высокое значение. 0: указывает, что система SoC может инициировать операцию ZQCS; 1: указывает, что операция ZQCS еще не была инициирована в uMCTL2. Режим программирования: динамический.	0x0	R
[5]	ctrlupd_busy	Система SoC может инициировать операцию ctrlupd только если этот сигнал имеет низкое значение. Этот сигнал получает высокое значение спустя такт после того, как uMCTL2 принимает запрос на ctrlupd. Он получает низкое значение, когда операция ctrlupd инициирована в uMCTL2. Рекомендуется не производить операций ctrlupd, когда этот сигнал имеет высокое значение. 0: указывает, что система SoC может инициировать операцию ctrlupd; 1: указывает, что операция ctrlupd еще не была инициирована в uMCTL2. Режим программирования: динамический.	0x0	R
[31:6]	–	Зарезервировано.	0	

24.6.106 Подключение настроек программирования регистров через программное обеспечение (SWCTL)

Поля регистра SWCTL описаны в Таблица 24.108.

Таблица 24.108. Подключение настроек программирования регистров через программное обеспечение

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	sw_done	Подключает внешний сброс квазидинамического программирования регистров. Следует запрограммировать в это поле 0, чтобы подключить квазидинамическое программирование. Следует установить в него обратно 1, после того как процесс программирования завершен. Режим программирования: динамический.	0x1	R/W
[31:1]	–	Зарезервировано.	0	

24.6.107 Статус настроек программирования регистров через программное обеспечение (SWSTAT)

Поля регистра SWSTAT описаны в Таблица 24.109.

Таблица 24.109. Статус настроек программирования регистров через программное обеспечение

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	sw_done_ack	Это поле указывает, что программирование регистров завершено. Это поле является отображением SWCTL.sw_done. Следует дождаться передачи значения 1 поля sw_done к полю sw_done_ack в конце последовательности программирования, чтобы убедиться, что к целевым доменам синхросигналов (destination clock domains) передались верные значения регистров. Контролепригодность: нетестируемо. Режим программирования: статический.	0x1	R
[31:1]	–	Зарезервировано.	0	

24.6.108 Регистр конфигурации сбоев AXI (POISONCFG)

Общий для всех портов интерфейса AXI.

Поля регистра POISONCFG описаны в Таблица 24.110.

Таблица 24.110. Регистр конфигурации сбоев AXI

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	wr_poison_slverr_en	Если в это поле установлено "1", подключает ответ SLVERR для сбоя транзакции записи (write transaction poisoning). Режим программирования: динамический.	0x1	R/W
[3:1]	–	Зарезервировано.	0	

Диапазон	Название	Описание	Исходное значение	Тип доступа
[4]	wr_poison_intr_en	Если в это поле установлено "1", подключает прерывания для сбоя транзакции записи. Режим программирования: динамический.	0x1	R/W
[7:5]	–	Зарезервировано.	0	
[8]	wr_poison_intr_clr	Очистка прерываний для сбоя транзакции записи. Отмеряет 2/3 тактов синхросигнала на то, чтобы правильное значение прошло к логике системы и очистило прерывания. uMCTL2 очищает этот бит автоматически. Контролепригодность: readOnly. Режим программирования: динамический.	0x0	R/W
[15:9]	–	Зарезервировано.	0	
[16]	rd_poison_slverr_en	Если в это поле установлено "1", подключает ответ SLVERR для сбоя транзакции чтения. Режим программирования: динамический.	0x1	R/W
[19:17]	–	Зарезервировано.	0	
[20]	rd_poison_intr_en	Если в это поле установлено "1", подключает прерывания для сбоя транзакции чтения. Режим программирования: динамический.	0x1	R/W
[23:21]	–	Зарезервировано.	0	
[24]	rd_poison_intr_clr	Очистка прерываний для сбоя транзакции чтения. Отмеряет 2/3 тактов синхросигнала на то, чтобы правильное значение прошло к логике системы и очистило прерывания. uMCTL2 очищает этот бит автоматически. Контролепригодность: readOnly. Режим программирования: динамический.	0x0	R/W
[31:25]	–	Зарезервировано.	0	

24.6.109 Регистр статуса сбоев AXI (POISONSTAT)

Поля регистра POISONSTAT описаны в Таблица 24.111.

Таблица 24.111. Регистр статуса сбоев AXI

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	wr_poison_intr_0	Прерывание ошибки о сбое транзакции записи для порта 0. Этот регистр - копия прерывания от синхросигнала расширенной периферийной шины APB (двойного синхронизатора регистров), которое устанавливается, когда обнаруживается сбой транзакции на канале адреса записи соответствующего порта AXI. Бит 0 соответствует порту 0 и так далее. Прерывание очищается полем wr_poison_intr_clr, затем значение передается синхросигналу APB. Режим программирования: динамический.	0x0	R
[1]	wr_poison_intr_1	Прерывание ошибки о сбое транзакции записи для порта 1. Этот регистр - копия прерывания от	0x0	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
		синхросигнала расширенной периферийной шины APB (двойного синхронизатора регистров), которое устанавливается, когда обнаруживается сбой транзакции на канале адреса записи соответствующего порта AXI. Бит 0 соответствует порту 0 и так далее. Прерывание очищается полем wr_poison_intr_clr, затем значение передается синхросигналу APB. Режим программирования: динамический.		
[2]	wr_poison_intr_2	Прерывание ошибки о сбое транзакции записи для порта 2. Этот регистр - копия прерывания от синхросигнала расширенной периферийной шины APB (двойного синхронизатора регистров), которое устанавливается, когда обнаруживается сбой транзакции на канале адреса записи соответствующего порта AXI. Бит 0 соответствует порту 0 и так далее. Прерывание очищается полем wr_poison_intr_clr, затем значение передается синхросигналу APB. Режим программирования: динамический.	0x0	R
[3]	wr_poison_intr_3	Прерывание ошибки о сбое транзакции записи для порта 3. Этот регистр - копия прерывания от синхросигнала расширенной периферийной шины APB (двойного синхронизатора регистров), которое устанавливается, когда обнаруживается сбой транзакции на канале адреса записи соответствующего порта AXI. Бит 0 соответствует порту 0 и так далее. Прерывание очищается полем wr_poison_intr_clr, затем значение передается синхросигналу APB. Режим программирования: динамический.	0x0	R
[4]	wr_poison_intr_4	Прерывание ошибки о сбое транзакции записи для порта 4. Этот регистр - копия прерывания от синхросигнала расширенной периферийной шины APB (двойного синхронизатора регистров), которое устанавливается, когда обнаруживается сбой транзакции на канале адреса записи соответствующего порта AXI. Бит 0 соответствует порту 0 и так далее. Прерывание очищается полем wr_poison_intr_clr, затем значение передается синхросигналу APB. Режим программирования: динамический.	0x0	R
[5]	wr_poison_intr_5	Прерывание ошибки о сбое транзакции записи для порта 5. Этот регистр - копия прерывания от синхросигнала расширенной периферийной шины APB (двойного синхронизатора регистров), которое устанавливается, когда обнаруживается сбой транзакции на канале адреса записи соответствующего порта AXI. Бит 0 соответствует порту 0 и так далее. Прерывание очищается полем wr_poison_intr_clr, затем значение передается синхросигналу APB. Режим программирования: динамический.	0x0	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
[6]	wr_poison_intr_6	Прерывание ошибки о сбое транзакции записи для порта 5. Этот регистр - копия прерывания от синхросигнала расширенной периферийной шины APB (двойного синхронизатора регистров), которое устанавливается, когда обнаруживается сбой транзакции на канале адреса записи соответствующего порта AXI. Бит 0 соответствует порту 0 и так далее. Прерывание очищается полем wr_poison_intr_clr, затем значение передается синхросигналу APB. Режим программирования: динамический.	0x0	R
[7]	wr_poison_intr_7	Прерывание ошибки о сбое транзакции записи для порта 7. Этот регистр - копия прерывания от синхросигнала расширенной периферийной шины APB (двойного синхронизатора регистров), которое устанавливается, когда обнаруживается сбой транзакции на канале адреса записи соответствующего порта AXI. Бит 0 соответствует порту 0 и так далее. Прерывание очищается полем wr_poison_intr_clr, затем значение передается синхросигналу APB. Режим программирования: динамический.	0x0	R
[8]	wr_poison_intr_8	Прерывание ошибки о сбое транзакции записи для порта 8. Этот регистр - копия прерывания от синхросигнала расширенной периферийной шины APB (двойного синхронизатора регистров), которое устанавливается, когда обнаруживается сбой транзакции на канале адреса записи соответствующего порта AXI. Бит 0 соответствует порту 0 и так далее. Прерывание очищается полем wr_poison_intr_clr, затем значение передается синхросигналу APB. Режим программирования: динамический.	0x0	R
[9]	wr_poison_intr_9	Прерывание ошибки о сбое транзакции записи для порта 9. Этот регистр - копия прерывания от синхросигнала расширенной периферийной шины APB (двойного синхронизатора регистров), которое устанавливается, когда обнаруживается сбой транзакции на канале адреса записи соответствующего порта AXI. Бит 0 соответствует порту 0 и так далее. Прерывание очищается полем wr_poison_intr_clr, затем значение передается синхросигналу APB. Режим программирования: динамический.	0x0	R
[10]	wr_poison_intr_10	Прерывание ошибки о сбое транзакции записи для порта 10. Этот регистр - копия прерывания от синхросигнала расширенной периферийной шины APB (двойного синхронизатора регистров),, которое устанавливается, когда обнаруживается сбой транзакции на канале адреса записи соответствующего порта AXI. Бит 0 соответствует порту 0 и так далее. Прерывание очищается полем wr_poison_intr_clr, затем значение передается синхросигналу APB.	0x0	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Режим программирования: динамический.		
[15:11]	–	Зарезервировано.	0	
[16]	rd_poison_intr_0	Прерывание ошибки о сбое транзакции чтения для порта 0. Этот регистр - копия прерывания от синхросигнала расширенной периферийной шины APB (двойного синхронизатора регистров), , которое устанавливается, когда обнаруживается сбой транзакции на канале адреса чтения соответствующего порта AXI. Бит 0 соответствует порту 0 и так далее. Прерывание очищается полем rd_poison_intr_clr, затем значение передается синхросигналу APB. Режим программирования: динамический.	0x0	R
[17]	rd_poison_intr_1	Прерывание ошибки о сбое транзакции чтения для порта 1. Этот регистр - копия прерывания от синхросигнала расширенной периферийной шины APB (двойного синхронизатора регистров), которое устанавливается, когда обнаруживается сбой транзакции на канале адреса чтения соответствующего порта AXI. Бит 0 соответствует порту 0 и так далее. Прерывание очищается полем rd_poison_intr_clr, затем значение передается синхросигналу APB. Режим программирования: динамический.	0x0	R
[18]	rd_poison_intr_2	Прерывание ошибки о сбое транзакции чтения для порта 2. Этот регистр - копия прерывания от синхросигнала расширенной периферийной шины APB (двойного синхронизатора регистров), которое устанавливается, когда обнаруживается сбой транзакции на канале адреса чтения соответствующего порта AXI. Бит 0 соответствует порту 0 и так далее. Прерывание очищается полем rd_poison_intr_clr, затем значение передается синхросигналу APB. Режим программирования: динамический.	0x0	R
[19]	rd_poison_intr_3	Прерывание ошибки о сбое транзакции чтения для порта 3. Этот регистр - копия прерывания от синхросигнала расширенной периферийной шины APB (двойного синхронизатора регистров), которое устанавливается, когда обнаруживается сбой транзакции на канале адреса чтения соответствующего порта AXI. Бит 0 соответствует порту 0 и так далее. Прерывание очищается полем rd_poison_intr_clr, затем значение передается синхросигналу APB. Режим программирования: динамический.	0x0	R
[20]	rd_poison_intr_4	Прерывание ошибки о сбое транзакции чтения для порта 4. Этот регистр - копия прерывания от синхросигнала расширенной периферийной шины APB (двойного синхронизатора регистров), которое устанавливается, когда обнаруживается сбой транзакции на канале адреса чтения соответствующего порта AXI. Бит 0 соответствует	0x0	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
		порту 0 и так далее. Прерывание очищается полем d_poison_intr_clr, затем значение передается синхросигналу APB. Режим программирования: динамический.		
[21]	rd_poison_intr_5	Прерывание ошибки о сбое транзакции чтения для порта 5. Этот регистр - копия прерывания от синхросигнала расширенной периферийной шины APB (двойного синхронизатора регистров), которое устанавливается, когда обнаруживается сбой транзакции на канале адреса чтения соответствующего порта AXI. Бит 0 соответствует порту 0 и так далее. Прерывание очищается полем rd_poison_intr_clr, затем значение передается синхросигналу APB. Режим программирования: динамический.	0x0	R
[22]	rd_poison_intr_6	Прерывание ошибки о сбое транзакции чтения для порта 6. Этот регистр - копия прерывания от синхросигнала расширенной периферийной шины APB (двойного синхронизатора регистров), которое устанавливается, когда обнаруживается сбой транзакции на канале адреса чтения соответствующего порта AXI. Бит 0 соответствует порту 0 и так далее. Прерывание очищается полем rd_poison_intr_clr, затем значение передается синхросигналу APB. Режим программирования: динамический.	0x0	R
[23]	rd_poison_intr_7	Прерывание ошибки о сбое транзакции чтения для порта 7. Этот регистр - копия прерывания от синхросигнала расширенной периферийной шины APB (двойного синхронизатора регистров), которое устанавливается, когда обнаруживается сбой транзакции на канале адреса чтения соответствующего порта AXI. Бит 0 соответствует порту 0 и так далее. Прерывание очищается полем rd_poison_intr_clr, затем значение передается синхросигналу APB. Режим программирования: динамический.	0x0	R
[24]	rd_poison_intr_8	Прерывание ошибки о сбое транзакции чтения для порта 8. Этот регистр - копия прерывания от синхросигнала расширенной периферийной шины APB (двойного синхронизатора регистров), которое устанавливается, когда обнаруживается сбой транзакции на канале адреса чтения соответствующего порта AXI. Бит 0 соответствует порту 0 и так далее. Прерывание очищается полем rd_poison_intr_clr, затем значение передается синхросигналу APB. Режим программирования: динамический.	0x0	R
[25]	rd_poison_intr_9	Прерывание ошибки о сбое транзакции чтения для порта 9. Этот регистр - копия прерывания от синхросигнала расширенной периферийной шины APB (двойного синхронизатора регистров), которое устанавливается, когда обнаруживается сбой	0x0	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
		транзакции на канале адреса чтения соответствующего порта AXI. Бит 0 соответствует порту 0 и так далее. Прерывание очищается полем rd_poison_intr_clr, затем значение передается синхросигналу APB. Режим программирования: динамический.		
[26]	rd_poison_intr_10	Прерывание ошибки о сбое транзакции чтения для порта 10. Этот регистр - копия прерывания от синхросигнала расширенной периферийной шины APB (двойного синхронизатора регистров), которое устанавливается, когда обнаруживается сбой транзакции на канале адреса чтения соответствующего порта AXI. Бит 0 соответствует порту 0 и так далее. Прерывание очищается полем rd_poison_intr_clr, затем значение передается синхросигналу APB. Режим программирования: динамический.	0x0	R
[31:27]	–	Зарезервировано.	0	

24.6.110 Расширенный регистр индексов ECC (ADVECCINDEX)

Поля регистра ADVECCINDEX описаны в Таблица 24.116

Таблица 24.112. Регистр ADVECCINDEX

Диапазон	Название	Описание	Исходное значение	Тип доступа
[2:0]	ecc_syndrome_sel	Регистр выбирает, какие передачи DRAM будут записываться в ECCCSYN0/1/2 и ECCUSYN0/1/2. Режим программирования: квазидинамическая группа 1.	0x0	R/W
[4:3]	ecc_err_symbol_sel	Регистр выбирает, какие ошибочные символы оказываются в ADVECCSTAT.advecc_err_symbol_pos и advecc_err_symbol_bits. По умолчанию записывается первый ошибочный символ. Значение должно быть меньше, чем ADVECCSTAT.advecc_num_err_symbol. Режим программирования: квазидинамическая группа 1.	0x0	R/W
[8:5]	ecc_poison_beats_sel	Регистр указывает, какие передачи в DRAM будут испорчены с помощью ECCPOISONPAT0/1/2. 4 передачи DRAM могут быть испорчены. Следует задать ecc_poison_beats_sel от 0 до 3, чтобы выбрать соответсвующую передачу (от 1-ой до 4-ой). Режим программирования: квазидинамическая группа 1.	0x0	R/W
[31:9]	–	Зарезервировано.	0	R

24.6.111 Регистр шаблона внесенной ошибки 0 (ECCPOISONPAT0)

Поля регистра ECCPOISONPAT0 описаны в Таблица 24.116

Таблица 24.113. Регистр ECCPOISONPAT0

Диапазон	Название	Описание	Исходное значение	Тип доступа
[31:0]	ecc_poison_data_31_0	Шаблон испорченных данных [31:0]. Установленные биты указывают на внесение ошибки (инверсию) в соответствующий бит DRAM. Режим программирования: квазидинамическая группа 3.	0x0	R/W

24.6.112 Регистр шаблона внесенной ошибки 2 (ECCPOISONPAT2)

Поля регистра ECCPOISONPAT2 описаны в Таблица 24.116

Таблица 24.114. Регистр ECCPOISONPAT2

Диапазон	Название	Описание	Исходное значение	Тип доступа
[7:0]	ecc_poison_data_71_64	Шаблон испорченных данных [71:64]. Установленные биты указывают на внесение ошибки (инверсию) в соответствующий бит DRAM. Режим программирования: квазидинамическая группа 3.	0x0	R/W
[31:8]	–	Зарезервировано	0	R

24.6.113 Регистр статуса защиты адреса (ECCAPSTAT)

Поля регистра ECCAPSTAT описаны в Таблица 24.116

Таблица 24.115. Регистр ECCAPSTAT

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	ecc_ap_err	Указывает превышение количества ошибок ECC (исправимых/неисправимых) в одной посылке значения ECCCFG0.ecc_ap_err_threshold. Режим программирования: динамический.	0x0	R
[31:1]	–	Зарезервировано.	0	R

24.6.114 Регистр статуса снижения максимально допустимых значений температуры (DERATESTAT)

Поля регистра DERATESTAT описаны в Таблица 24.116

Таблица 24.116. Регистр статуса снижения максимально допустимых значений температуры

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	derate_temp_limit_intr	Прерывание снижения температуры, указывающее, что предел рабочей температуры LPDDR2/3/4 SDRAM превышен. Это поле получает значение 1, когда значение, читаемое из MR4[2:0], равно 3'b000 или 3'b111. Это поле очищается регистром DERATECTL.derate_temp_limit_intr_clr.	0x0	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Контролепригодность: readOnly. Режим программирования: статический.		
[31:1]	–	Зарезервировано.	0	

24.7 Регистры блока UMCTL2_MP

Карта регистров блока UMCTL2_MP приведена в Таблица 24.2.

Таблица 24.117. Карта регистров UMCTL2_MP

№	Регистр	Описание	Смещение
1	PSTAT	Регистр статуса портов	0x3fc
2	PCCFG	Регистр общей конфигурации портов	0x400
3	PCFGR_0	Регистр конфигурирования чтений порта n	0x404
4	PCFGW_0	Регистр конфигурирования записей порта n	0x408
5	PCTRL_0	Регистр управления портом n	0x490
6	PCFGQOS0_0	Регистр 0 конфигурирования качества обработки (QoS) чтений порта n	0x494
7	PCFGR_1	Регистр конфигурирования чтений порта n	0x4b4
8	PCFGW_1	Регистр конфигурирования записей порта n	0x4b8
9	PCTRL_1	Регистр управления портом n	0x540
10	PCFGQOS0_1	Регистр 0 конфигурирования качества обработки чтений порта n	0x544
11	PCFGR_2	Регистр конфигурирования чтений порта n	0x564
12	PCFGW_2	Регистр конфигурирования записей порта n	0x568
13	PCTRL_2	Регистр управления портом n	0x5f0
14	PCFGQOS0_2	Регистр 0 конфигурирования качества обработки чтений порта n	0x5f4
15	PCFGR_3	Регистр конфигурирования чтений порта n	0x614
16	PCFGW_3	Регистр конфигурирования записей порта n	0x618
17	PCTRL_3	Регистр управления портом n	0x6a0

№	Регистр	Описание	Смещение
18	PCFGQOS0_3	Регистр 0 конфигурирования качества обработки чтений порта n	0x6a4
19	PCFGR_4	Регистр конфигурирования чтений порта n	0x6c4
20	PCFGW_4	Регистр конфигурирования записей порта n	0x6c8
21	PCTRL_4	Регистр управления портом n	0x750
22	PCFGQOS0_4	Регистр 0 конфигурирования качества обработки чтений порта n	0x754
23	PCFGR_5	Регистр конфигурирования чтений порта n	0x774
24	PCFGW_5	Регистр конфигурирования записей порта n	0x778
25	PCTRL_5	Регистр управления портом n	0x800
26	PCFGQOS0_5	Регистр 0 конфигурирования качества обработки чтений порта n	0x804
27	PCFGR_6	Регистр конфигурирования чтений порта n	0x824
28	PCFGW_6	Регистр конфигурирования записей порта n	0x828
29	PCTRL_6	Регистр управления портом n	0x8b0
30	PCFGQOS0_6	Регистр 0 конфигурирования качества обработки чтений порта n	0x8b4
31	PCFGR_7	Регистр конфигурирования чтений порта n	0x8d4
32	PCFGW_7	Регистр конфигурирования записей порта n	0x8d8
33	PCTRL_7	Регистр управления портом n	0x960
34	PCFGQOS0_7	Регистр 0 конфигурирования качества обработки чтений порта n	0x964
35	PCFGR_8	Регистр конфигурирования чтений порта n	0x984
36	PCFGW_8	Регистр конфигурирования записей порта n	0x988
37	PCTRL_8	Регистр управления портом n	0xa10
38	PCFGQOS0_8	Регистр 0 конфигурирования качества обработки чтений порта n	0xa14
39	PCFGR_9	Регистр конфигурирования чтений порта n	0xa34
40	PCFGW_9	Регистр конфигурирования записей порта n	0xa38

№	Регистр	Описание	Смещение
41	PCTRL_9	Регистр управления портом n	0xac0
42	PCFGQOS0_9	Регистр 0 конфигурирования качества обработки чтений порта n	0xac4
43	PCFGR_10	Регистр конфигурирования чтений порта n	0xae4
44	PCFGW_10	Регистр конфигурирования записей порта n	0xae8
45	PCTRL_10	Регистр управления портом n	0xb70
46	PCFGQOS0_10	Регистр 0 конфигурирования качества обработки чтений порта n	0xb74
47	SARBASE0	Регистр 0 мэппинга системных адресов	0xf04
48	SARSIZE0	Регистр 0 размера мэппинга системных адресов	0xf08
49	SARBASE1	Регистр 1 мэппинга системных адресов	0xf0c
50	SARSIZE1	Регистр 1 размера мэппинга системных адресов	0xf10
51	SARBASE2	Регистр 2 мэппинга системных адресов	0xf14
52	SARSIZE2	Регистр 2 размера мэппинга системных адресов	0xf18
53	SARBASE3	Регистр 3 мэппинга системных адресов	0xf1c
54	SARSIZE3	Регистр 3 размера мэппинга системных адресов	0xf20
55	UMCTL2_VER_NUMBER	Регистр версии контроллера (UMCTL2 Version Number Register)	0xff0
56	UMCTL2_VER_TYPE	Регистр типа версии контроллера (UMCTL2 Version Type Register)	0xff4

24.7.1 Регистр статуса портов (PSTAT)

Поля регистра PSTAT описаны в Таблица 24.118.

Таблица 24.118. Регистр статуса портов

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	rd_port_busy_0	Указывает, присутствуют ли необработанные команды чтения для порта AXI 0. Режим программирования: динамический.	0x0	R
[1]	rd_port_busy_1	Указывает, присутствуют ли необработанные команды чтения для порта AXI 1. Режим программирования: динамический.	0x0	R
[2]	rd_port_busy_2	Указывает, присутствуют ли необработанные команды чтения для порта AXI 2. Режим программирования: динамический.	0x0	R
[3]	rd_port_busy_3	Указывает, присутствуют ли необработанные команды чтения для порта AXI 3. Режим программирования: динамический.	0x0	R
[4]	rd_port_busy_4	Указывает, присутствуют ли необработанные команды чтения для порта AXI 4. Режим программирования: динамический.	0x0	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
[5]	rd_port_busy_5	Указывает, присутствуют ли необработанные команды чтения для порта AXI 5. Режим программирования: динамический.	0x0	R
[6]	rd_port_busy_6	Указывает, присутствуют ли необработанные команды чтения для порта AXI 6. Режим программирования: динамический.	0x0	R
[7]	rd_port_busy_7	Указывает, присутствуют ли необработанные команды чтения для порта AXI 7. Режим программирования: динамический.	0x0	R
[8]	rd_port_busy_8	Указывает, присутствуют ли необработанные команды чтения для порта AXI 8. Режим программирования: динамический.	0x0	R
[9]	rd_port_busy_9	Указывает, присутствуют ли необработанные команды чтения для порта AXI 9. Режим программирования: динамический.	0x0	R
[10]	rd_port_busy_10	Указывает, присутствуют ли необработанные команды чтения для порта AXI 10. Режим программирования: динамический.	0x0	R
[15:11]	–	Зарезервировано.	0	
[16]	wr_port_busy_0	Указывает, присутствуют ли необработанные команды записи для порта AXI 0. Режим программирования: динамический.	0x0	R
[17]	wr_port_busy_1	Указывает, присутствуют ли необработанные команды записи для порта AXI 1. Режим программирования: динамический.	0x0	R
[18]	wr_port_busy_2	Указывает, присутствуют ли необработанные команды записи для порта AXI 2. Режим программирования: динамический.	0x0	R
[19]	wr_port_busy_3	Указывает, присутствуют ли необработанные команды записи для порта AXI 3. Режим программирования: динамический.	0x0	R
[20]	wr_port_busy_4	Указывает, присутствуют ли необработанные команды записи для порта AXI 4. Режим программирования: динамический.	0x0	R
[21]	wr_port_busy_5	Указывает, присутствуют ли необработанные команды записи для порта AXI 5. Режим программирования: динамический.	0x0	R
[22]	wr_port_busy_6	Указывает, присутствуют ли необработанные команды записи для порта AXI 6. Режим программирования: динамический.	0x0	R
[23]	wr_port_busy_7	Указывает, присутствуют ли необработанные команды записи для порта AXI 7. Режим программирования: динамический.	0x0	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
[24]	wr_port_busy_8	Указывает, присутствуют ли необработанные команды записи для порта AXI 8. Режим программирования: динамический.	0x0	R
[25]	wr_port_busy_9	Указывает, присутствуют ли необработанные команды записи для порта AXI 9. Режим программирования: динамический.	0x0	R
[26]	wr_port_busy_10	Указывает, присутствуют ли необработанные команды записи для порта AXI 10. Режим программирования: динамический.	0x0	R
[31:27]	–	Зарезервировано.	0	

24.7.2 Регистр общей конфигурации портов (PCCFG)

Поля регистра PCCFG описаны в Таблица 24.119.

Таблица 24.119. Регистр общей конфигурации портов

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	go2critical_en	Если установлена 1 (бит подключен), устанавливает сигналы co_gs_go2critical_wr и co_gs_go2critical_lpr/co_gs_go2critical_hpr, идущие к DDRC, на основе первоочередных данных ввода (awurgent, arurgent) от главного интерфейса AXI (AXI master). Если установлен 0 (бит отключен), сигналы co_gs_go2critical_wr и co_gs_go2critical_lpr/co_gs_go2critical_hpr на DDRC получают значение 1b'0. Режим программирования: статический.	0x0	R/W
[3:1]	–	Зарезервировано.	0	
[4]	pagematch_limit	Лимит на 4 совпадения страниц (Page match four limit). Если установлена 1, ограничивает количество идущих подряд транзакций DDRC к одной и той же странице, которое может быть разрешено арбитром порта (Port Arbiter), до четырех, при условии, что функция совпадения страниц (Page Match feature) подключена. Если установлен 0, лимит на число идущих подряд транзакций DDRC к одной и той же странице не задан. Режим программирования: статический.	0x0	R/W
[7:5]	–	Зарезервировано.	0	
[8]	bl_exp_mode	Режим расширения длины импульса (Burst length expansion mode). По умолчанию (например, bl_exp_mode==0) XPI (программный интерфейс выхода) расширяет каждый импульс AXI во множество команд HIF, используя длину импульса (пакета) памяти как единицу измерения.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Если установлена 1, тогда XPI задействует половину длины импульса памяти как единицу измерения. Это применимо как к чтениям, так и к записям. Когда MSTR.data_bus_width==00, установка 1 в bl_exp_mode ни на что не влияет. Это может быть использовано в случаях, когда подключены частичные записи (Partial Writes, UMCTL2_PARTIAL_WR=1), чтобы избежать или минимизировать потери на задержку t_ccd_l в DDR4 и задержку t_ccd_mw в LPDDR4. Таким образом, рекомендуется устанавливать bl_exp_mode=1 только при использовании DDR4 или LPDDR4. Следует отметить, что если DBICTL.dm_en=0, то в следующих случаях функционал не поддерживается: - UMCTL2_PARTIAL_WR=0 UMCTL2_PARTIAL_WR=1, MSTR.data_bus_width=01, MEMC_BURST_LENGTH=8 и MSTR.burst_rdwr=1000 (только для LPDDR4); - UMCTL2_PARTIAL_WR=1, MSTR.data_bus_width=01, MEMC_BURST_LENGTH=4 и MSTR.burst_rdwr=0100 (только для DDR4), при этом либо MSTR.burstchop=0, либо CRCPARCTL1.crc_enable=1. Функционал также не поддерживается если подключено чередование каналов данных (Data Channel Interleave). Режим программирования: статический.		
[31:9]	–	Зарезервировано.	0	

24.7.3 Регистр конфигурирования чтений порта n (PCFGR_0 - PCFGR_10)

Поля регистров PCFGR_0 - PCFGR_10 описаны в Таблица 24.120.

Таблица 24.120. Регистр конфигурирования чтений порта n

Диапазон	Название	Описание	Исходное значение	Тип доступа
[9:0]	rd_port_priority	Определяет значение первичной загрузки счетчиков устаревания чтения (read aging counters). Эти счетчики будут загружены параллельно после сброса, или после каждого допуска к соответствующему порту. Счетчики устаревания ведут обратный отсчёт во всякий такт синхросигнала, когда запрос от порта произведен, но допуск не дан. Старшие 5 битов счетчика устаревания чтений задают приоритет канала чтения данного порта. Приоритет порта будет увеличиваться с уменьшением значения старших 5-ти битов счетчика. Когда счётчик устаревания станет нулевым, канал соответствующего порта получит	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		наивысший уровень приоритета (условие таймаута - Priority0). В многопортовых конфигурациях счетчики устаревания не могут использоваться для установки приоритета портов, когда подключены внешние вводы динамического приоритета (arqos) (условие таймаута по-прежнему применимо). В однопортовых конфигурациях счетчики устаревания применяются только когда они уходят в таймаут (обнуляются), для принудительной установки переключения направлений чтения-записи. В этом случае, внешний ввод динамического приоритета, arqos (только для чтений) по-прежнему может быть использован для установки приоритета чтения DDRC (2 уровня приоритета: низкоприоритетное чтение - LPR, высокоприоритетное чтение - HPR) в формате "команда за командой" (command by command). Замечание: два младших бита этого поля внутренне привязаны к значению 2'b00. Режим программирования: статический.		
[11:10]	–	Зарезервировано.	0	
[12]	rd_port_aging_en	Если установлена 1, подключает функцию устаревания (aging function) для канала чтения данного порта. Режим программирования: статический.	0x0	R/W
[13]	rd_port_urgent_en	Если установлена 1, подключает первоочередной сигнал AXI с одной боковой полосой (arurgent). Когда этот бит подключен и arurgent установлен через главное устройство (master), этот порт получает наивысший приоритет и, если подключено поле PCCFG.go2critical_en, устанавливается сигнал co_gs_go2critical_lpr/co_gs_go2critical_hpr к DDRC. Следует отметить, что сигнал arurgent может быть установлен в любой момент на столько, на сколько необходимо, что не зависит от согласования адресов (address handshaking) (он не связан ни с какой определенной командой). Режим программирования: статический.	0x0	R/W
[14]	rd_port_pagematch_en	Если установлена 1, подключает функцию совпадения страницы. Если она подключена, то в момент, когда запрашивающему порту выдается допуск, если команды, следующие сразу после этого, относятся к одной и той же странице памяти (тому же банку и той же строке), выдача допуска порту продолжается. Также с этой функцией связано поле PCCFG.pagematch_limit. Режим программирования: статический.	0x0	R/W
[31:15]	–	Зарезервировано.	0	

24.7.4 Регистр конфигурирования записей порта n (PCFGW_0 - PCFGW_10)

Поля регистра PCFGW_0 - PCFGW_10 описаны в Таблица 24.121.

Таблица 24.121. Регистр конфигурирования записей порта n

Диапазон	Название	Описание	Исходное значение	Тип доступа
[9:0]	wr_port_priority	Определяет значение первичной загрузки счетчиков устаревания записи (write aging counters). Эти счетчики будут загружены параллельно после сброса, или после каждого допуска к соответствующему порту. Счетчики устаревания ведут обратный отсчёт во всякий такт синхросигнала, когда запрос от порта произведен, но допуск не дан. Старшие 5 битов счетчика устаревания записей задают приоритет канала записи данного порта. Приоритет порта будет увеличиваться с уменьшением значения старших 5-ти битов счетчика. Когда счётчик устаревания станет нулевым, канал соответствующего порта получит наивысший уровень приоритета (условие таймаута - Priority0). В многопортовых конфигурациях счетчики устаревания не могут использоваться для установки приоритета портов, когда подключены внешние вводы динамического приоритета (arqos) (условие таймаута по-прежнему применимо). В однопортовых конфигурациях счетчики устаревания применяются только когда они уходят в таймаут (обнуляются), для принудительной установки переключения направлений чтения-записи. В этом случае, внешний ввод динамического приоритета, arqos (только для чтений) по-прежнему может быть использован для установки приоритета чтения DDRC. Замечание: два младших бита этого поля внутренне привязаны к значению 2'b00. Режим программирования: статический.	0x0	R/W
[11:10]	–	Зарезервировано.	0	
[12]	wr_port_aging_en	Если установлена 1, подключает функцию устаревания (aging function) для канала записи данного порта. Режим программирования: статический.	0x0	R/W
[13]	wr_port_urgent_en	Если установлена 1, подключает первоочередной сигнал AXI с одной боковой полосой (awurgent). Когда этот бит подключен и awurgent установлен через главное устройство (master), этот порт получает наивысший приоритет и, если подключено поле PCCFG.go2critical_en, устанавливается сигнал co_gs_go2critical_wr к DDRC. Следует отметить, что сигнал awurgent может быть установлен в любой момент на столько, на сколько необходимо, что не зависит от согласования адресов	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		(address handshaking) (он не связан ни с какой определенной командой). Режим программирования: статический.		
[14]	wr_port_pagematch_en	Если установлена 1, подключает функцию совпадения страницы. Если она подключена, то в момент, когда запрашивающему порту выдается допуск, если команды, следующие сразу после этого, относятся к одной и той же странице памяти (тому же банку и той же строке), выдача допуска порту продолжается. Также с этой функцией связано поле PCCFG.pagematch_limit. Режим программирования: статический.	0x1	R/W
[31:15]	–	Зарезервировано.	0	

24.7.5 Регистр управления портом n (PCTRL_0 - PCTRL_10)

Поля регистра PCTRL_0 - PCTRL_10 описаны в Таблица 24.122.

Таблица 24.122. Регистр управления портом n

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	port_en	Подключает порт n интерфейса AXI. Режим программирования: динамический.	0x0	R/W
[31:1]	–	Зарезервировано.	0	

24.7.6 Регистр 0 конфигурирования качества обработки (QoS) чтений порта n (PCFGQOS0_0 - PCFGQOS0_10)

Поля регистра PCFGQOS0_0 - PCFGQOS0_10 описаны в Таблица 24.123.

Таблица 24.123. Регистр 0 конфигурирования качества обработки (QoS) чтений порта n

Диапазон	Название	Описание	Исходное значение	Тип доступа
[3:0]	rqos_map_level1	Функциональное разделение level1, указывающее конец соответствия region0; начало region0 - 0. Возможные значения для level1 - от 0 до 13 (для двойной очереди адресов чтения (dual RAQ)) или от 0 до 14 (для одинарной RAQ) в соответствии с arqos. Следует отметить, что для PA (арбитра портов), значения arqos напрямую используются как приоритеты портов, где большее значение соответствует большему приоритету порта. Во все регистры map_level* необходимо устанавливать несовпадающие значения. Режим программирования: квазидинамическая группа 3.	0x0	R/W
[15:4]	–	Зарезервировано.	0	

Диапазон	Название	Описание	Исходное значение	Тип доступа
[17:16]	rqos_map_region0	Это поле указывает класс траффика области region0. Валидные значения: 0: LPR, 1: VPR, 2: HPR. Для конфигураций двойной очереди адресов, region0 соответствует синей очереди адресов (примечание: в блоке XPI существуют две раздельные очереди адресов: красная и синяя. Синяя может использоваться для двух классов траффика). В этом случае, валидные значения: только 0: LPR и 1: VPR. Когда поддержка VPR отключена (UMCTL2_VPR_EN = 0) и класс траффика region0 имеет значение 1 (VPR), траффик VPR смешивается (aliased) с траффиком LPR. Режим программирования: квазидинамическая группа 3.	0x0	R/W
[19:18]	–	Зарезервировано.	0	
[21:20]	rqos_map_region1	Это поле указывает класс траффика области region1. Валидные значения: 0: LPR, 1: VPR, 2: HPR. Для конфигураций двойной очереди адресов, region1 соответствует синей очереди адресов (примечание: в блоке XPI существуют две раздельные очереди адресов: красная и синяя. Синяя может использоваться для двух классов траффика). В этом случае, валидные значения: только 0: LPR и 1: VPR. Когда поддержка VPR отключена (UMCTL2_VPR_EN = 0) и класс траффика region1 имеет значение 1 (VPR), траффик VPR смешивается (aliased) с траффиком LPR. Режим программирования: квазидинамическая группа 3.	0x0	R/W
[31:22]	–	Зарезервировано.	0	

24.7.7 Регистр 0 мэппинга системных адресов

Поля регистра SARBASE0 описаны в Таблица 24.123.

Таблица 24.124. Регистр 0 мэппинга системных адресов

Диапазон	Название	Описание	Исходное значение	Тип доступа
[11:0]	base_addr	Базовый адрес региона 0, задаваемый awaddr[39:28] и araddr[39:28]. Режим программирования: статический.	0x0	R/W
[31:12]	–	Зарезервировано.	0	

24.7.8 Регистр 0 размера мэппинга системных адресов

Поля регистра SARSIZE0 описаны в Таблица 24.123.

Таблица 24.125. Регистр 0 мэппинга системных адресов

Диапазон	Название	Описание	Исходное значение	Тип доступа
[7:0]	nblocks	Количество блоков размера 256 МБайт в регионе 0. Данный регистр определяет общий размер региона путем умножения количества заданных блоков (+1) умноженных на 256 МБайт. Режим программирования: статический.	0x0	R/W
[31:8]	–	Зарезервировано.	0	

24.7.9 Регистр 1 мэппинга системных адресов

Поля регистра SARBASE1 описаны в Таблица 24.123.

Таблица 24.126. Регистр 1 мэппинга системных адресов

Диапазон	Название	Описание	Исходное значение	Тип доступа
[11:0]	base_addr	Базовый адрес региона 1, задаваемый awaddr[39:28] и araddr[39:28]. Режим программирования: статический.	0x0	R/W
[31:12]	–	Зарезервировано.	0	

24.7.10 Регистр 1 размера мэппинга системных адресов

Поля регистра SARSIZE1 описаны в Таблица 24.123.

Таблица 24.127. Регистр 1 мэппинга системных адресов

Диапазон	Название	Описание	Исходное значение	Тип доступа
[7:0]	nblocks	Количество блоков размера 256 МБайт в регионе 1. Данный регистр определяет общий размер региона путем умножения количества заданных блоков (+1) умноженных на 256 МБайт. Режим программирования: статический.	0x0	R/W
[31:8]	–	Зарезервировано.	0	

24.7.11 Регистр 2 мэппинга системных адресов

Поля регистра SARBASE2 описаны в Таблица 24.123.

Таблица 24.128. Регистр 2 мэппинга системных адресов

Диапазон	Название	Описание	Исходное значение	Тип доступа
[11:0]	base_addr	Базовый адрес региона 2, задаваемый awaddr[39:28] и araddr[39:28]. Режим программирования: статический.	0x0	R/W
[31:12]	–	Зарезервировано.	0	

24.7.12 Регистр 2 размера мэппинга системных адресов

Поля регистра SARSIZE2 описаны в Таблица 24.123.

Таблица 24.129. Регистр 2 мэппинга системных адресов

Диапазон	Название	Описание	Исходное значение	Тип доступа
[7:0]	nblocks	Количество блоков размера 256 МБайт в регионе 2. Данный регистр определяет общий размер региона путем умножения количества заданных блоков (+1) умноженных на 256 МБайт. Режим программирования: статический.	0x0	R/W
[31:8]	–	Зарезервировано.	0	

24.7.13 Регистр 3 мэппинга системных адресов

Поля регистра SARBASE3 описаны в Таблица 24.123.

Таблица 24.130. Регистр 3 мэппинга системных адресов

Диапазон	Название	Описание	Исходное значение	Тип доступа
[11:0]	base_addr	Базовый адрес региона 3, задаваемый awaddr[39:28] и araddr[39:28]. Режим программирования: статический.	0x0	R/W
[31:12]	–	Зарезервировано.	0	

24.7.14 Регистр 3 размера мэппинга системных адресов

Поля регистра SARSIZE3 описаны в Таблица 24.123.

Таблица 24.131. Регистр 3 мэппинга системных адресов

Диапазон	Название	Описание	Исходное значение	Тип доступа
[7:0]	nblocks	Количество блоков размера 256 МБайт в регионе 3. Данный регистр определяет общий размер региона путем умножения количества заданных блоков (+1) умноженных на 256 МБайт. Режим программирования: статический.	0x0	R/W
[31:8]	–	Зарезервировано.	0	

24.7.15 Регистр версии контроллера

Поля регистра UMCTL2_VER_NUMBER описаны в Таблица 24.123.

Таблица 24.132. Регистр версии контроллера

Диапазон	Название	Описание	Исходное значение	Тип доступа
[31:0]	ver_number	Указывает номер версии устройства. Режим программирования: статический.	0x3333302a	R

24.7.16 Регистр типа версии контроллера

Поля регистра UMCTL2_VER_TYPE описаны в Таблица 24.123.

Таблица 24.133. Регистр типа версии контроллера

Диапазон	Название	Описание	Исходное значение	Тип доступа
[31:0]	ver_type	Указывает тип версии устройства. Режим программирования: статический.	0x67612a2a	R

24.8 Функционирование

24.8.1 Инициализация

Перед использованием контроллера DDR, PHY и динамической памяти, следует провести процедуру инициализации. Инициализация состоит из двух частей. Первая происходит автоматически после выхода схемы из состояния сброса:

1. PHY находится в неинициализированном состоянии в течение всего времени сброса (presetn равен 0).

2. При выходе из сброса PHY переходит в режим инициализации DLL. Данный режим может быть пропущен записью 1 в PIR[LOCKBYP].

3. Параллельно с инициализацией DLL запускается процедура калибровки сопротивлений (impedance calibration). Этот режим может быть пропущен записью 1 в PIR{ZCALBYP]. Подробности режима калибровки сопротивлений представлены в разделе «Калибровка сопротивлений».

4. Если процедура инициализации PHY была активирована программно, программный сброс может быть опционально выполнен для ITM. Это связано с тем, что в процессе аппаратной инициализации логика ITM сбрасывается основным сбросом, в то время как при программной инициализации такого сброса нет.

5. По окончании инициализации DLL, калибровки сопротивлений и сброса ITM PHY переводится в инициализированное состояние. В случае, если указанные процедуры были пропущены, пользователь должен убедиться, что они будут выполнены перед использованием PHY.

Вторая стадия инициализации начинается после инициализации PHY. Каждый этап управляется контроллером памяти.

1. Контроллер DDR (или PUBL) инициируют процедуру инициализации SDRAM. Регистры управления динамической памятью (mode registers) и временные параметры должны быть запрограммированы перед запуском процедуры инициализации SDRAM.

2. По окончании инициализации SDRAM требуется запустить процедуру тренинга положения маски стробов (DQS gate training).

3. Запустить процедуру тренинга смещения стробов (data valid training).

4. Запустить процедуру read data eye training (!!! В данной версии не поддерживается !!!).

5. PHY готов к записи/чтению динамической памяти.

24.8.2 Инициализация SDRAM

Замечание. Регистры PUBL, регистры SDRAM и регистры контроллера DDR должны быть запрограммированы эквивалентными значениями во избежание некорректного функционирования контроллера памяти.

DDR SDRAM должна быть проинициализирована перед исполнением команд записи/чтения. PUBL содержит встроенный автомат инициализации SDRAM, который может быть активирован записью в PIR. Рекомендуется использовать данный метод инициализации, однако в некоторых случаях может потребоваться проинициализировать память другим способом:

1. Следует дождаться окончания инициализации PHY.

2. Записать 1 в PIR[INIT]. Бит PIR[CTLDINIT] следует записать в соответствии с требуемым типом инициализации: 1 (инициализация контроллером DDR) / 0 (инициализация PUBL и программно).

3. Выполнить процедуру инициализации программно или с помощью контроллера DDR. Программная инициализация может быть выполнена с помощью DRAM Command Unit (DCU).

Процедуры инициализации для различных типов памяти отличаются.

Все этапы инициализации могут быть активированы с использованием PHY Initialization Register (PIR). Запись 1 в PIR[INIT] активирует инициализацию с использованием процедур, выбранных соответствующими битами PIR. Если выбрано несколько

процедур, они исполняются последовательно. Завершение процедур может быть отслежено в PHY General Status Register (PGSR). Бит PGSR[IDONE] сигнализирует окончание всех выбранных процедур в составе инициализации. Бит сбрасывается при активации любой из процедур в составе инициализации.

Замечание. Процедуры, которые были пропущены по указанию пользователя, помечаются как выполненные. Пользователь должен выполнить данные процедуры перед корректным использованием PHY.

25. КОНТРОЛЛЕРЫ PCI-EXPRESS (PCIE) И JESD204B

25.1 Общая информация

В главе описана работа контроллеров PCI-EXPRESS (PCIe) и Jesd204b. В системе установлены два контроллера PCIe и два контроллера Jesd204b. Контроллеры используют единый физический уровень и общий набор контактных площадок из-за этого их совместная работа невозможна (одновременно могут работать два контроллера PCIe или два контроллера Jesd204b или один контроллер PCIe и один контроллер Jesd204b). Выбор режима работы PCIe/Jesd204b осуществляется программно (по умолчанию выбран режим работы “PCIe”).

25.1.1 Контроллер PCI Express

Контроллер PCI Express 3.0 Dual Mode соответствует спецификации PCI Express Base Specification, Revision 4.0, Version 1.0.

25.1.1.1 Основные характеристики контроллера

- Поддержка работы в одном из режимов Root Complex или End Point.
- Прием и передача данных по четырем последовательным каналам, со скоростью до 8 Гбит/с для спецификации Gen 3.
- Содержит 32 области преобразования входящих адресов и 32 области преобразования исходящих адресов, размером от 4 KB до 1 TB.
- Встроенный DMA включающих в себя четыре канала чтения и четыре канала записи.
- Поддержка MSI, MSI-X и legacy сообщений прерываний (INTx). Со встроенным модулем генерации MSI-X.
- Четыре виртуальных канала.
- Конфигурация размера Max_Payload_Size от 128 байт до 4 Кбайт.
- Доступ к регистрам PHY.
- Поддержка автоматического реверса линий.

Таблица 25.1 Доступные BAR диапазоны

№	Разрядность	Тип доступа	Programmable mask	Маска (размер BAR)	Область доступа (пример распределения)	Возможность переопределения
BAR0	32-bit	memory	Yes	0x3fff_ffff	1 ГБ внутренняя память	Yes
BAR1	32-bit	memory	Yes	0x7fff_ffff	2 ГБ DDRlow	Yes
BAR2/BAR3	64-bit	memory	Yes	0x7_ffff_ffff	32 ГБ DDRhigh	Yes
BAR4	32-bit	memory	Yes	0xffff	MSI-X	No
BAR5	32-bit	memory	Yes	0xffff	DMA + ATU	No

25.1.2 Контроллер Jesd204b

Контроллер Jesd204b обеспечивает поддержку стандарта высокоскоростной последовательной связи, предназначенной для подключения аналого-цифрового преобразователя (АЦП) и цифро-аналогового преобразователя (ЦАП). Он обеспечивает линейную скорость до 12,5 Гбит/с на линию, гарантируя выравнивание и синхронизацию данных. Ядро обеспечивает функционирование как передатчика (TX), так и приемника (RX) и может динамически настраиваться для удовлетворения любых требований по выравниванию нескольких преобразователей, нескольких полос (MCDA-ML).

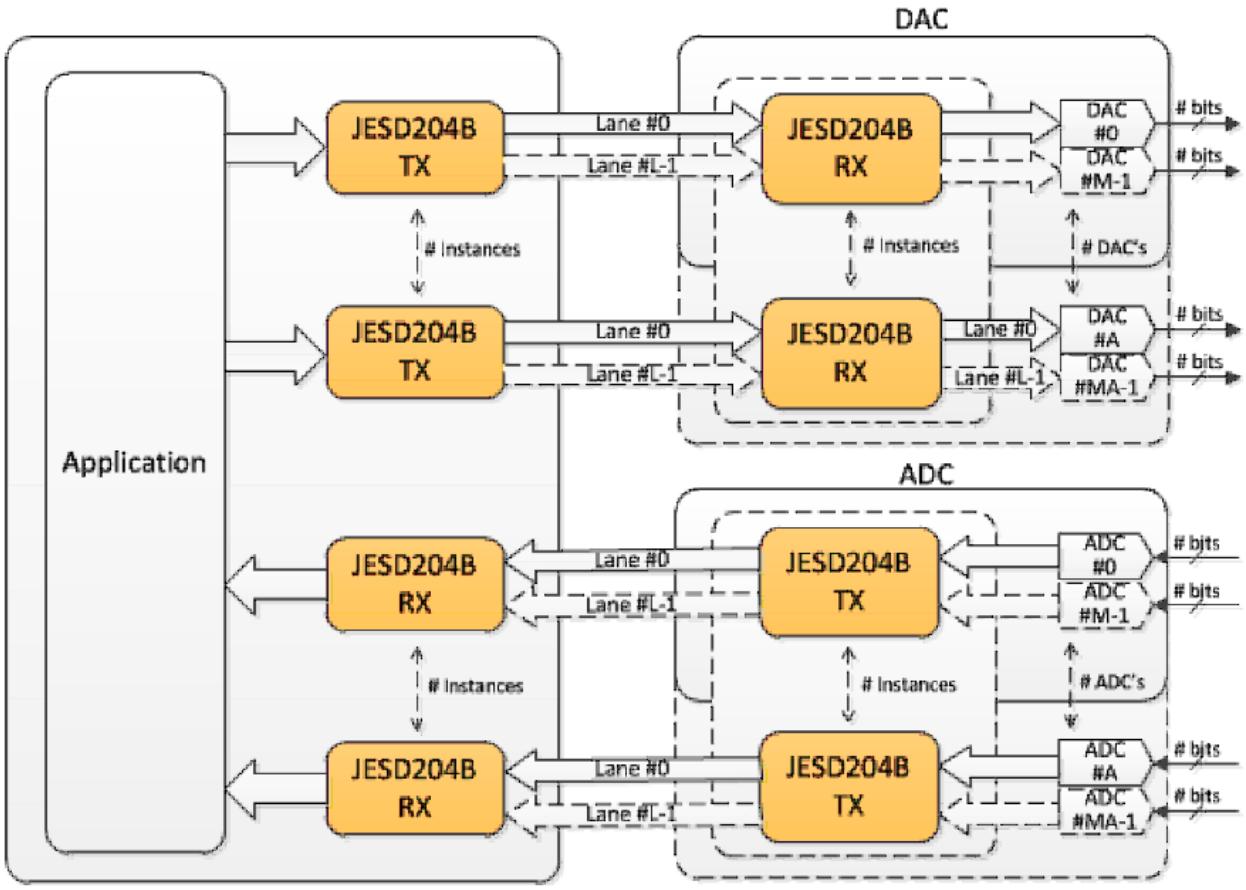


Рисунок 25.1. Схема подключения устройств по JESD204b протоколу

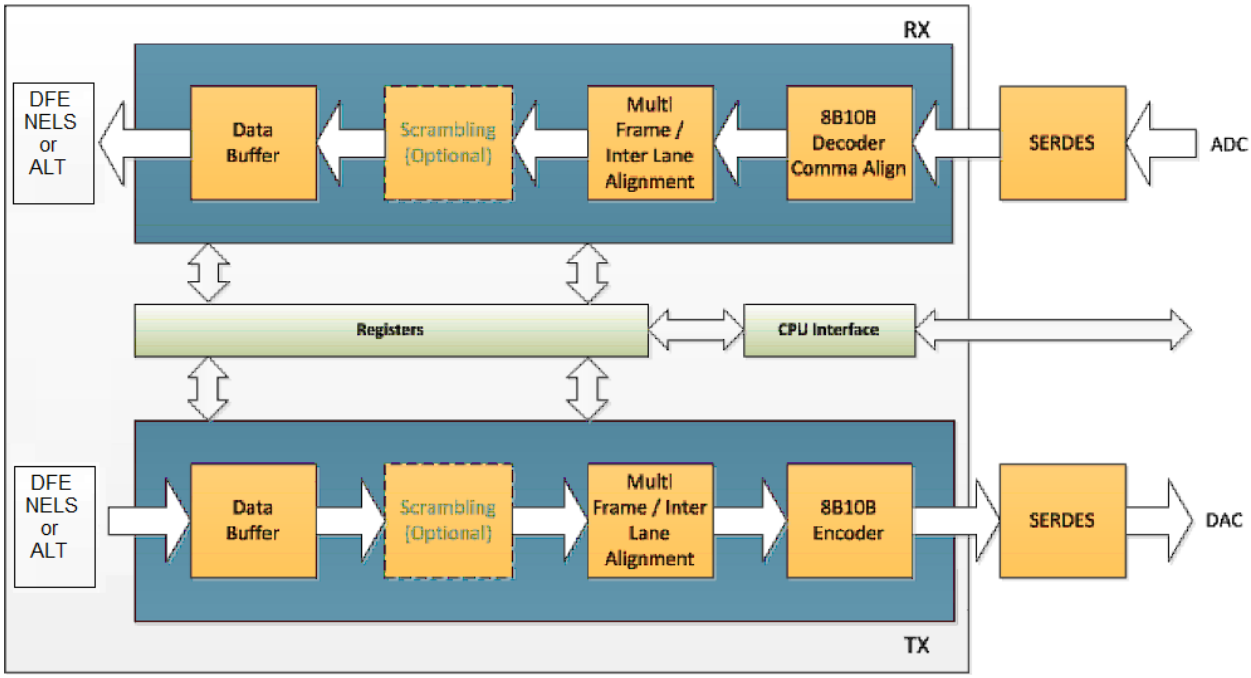


Рисунок 25.2. Структурная схема контроллера Jesd204b

25.1.2.1 Основные характеристики:

- полное соответствие стандарту JESD204B
    - поддержка подклассов 0, 1, 2
- скорость передачи до 12.5 Гбит/с на линию
- максимальное количество линий – 4
- максимальное количество конвертеров на устройство – 8
- разрешение конвертера – от 8 до 16
- количество бит в семпле – 8/12/16
- поддержка контрольного слова
- количество контрольных бит на семпл – от 0 до 3
- поддержка HD режима
- мультифреймы – от 1 до 32
- размер фрейма – от 1 до 8

25.1.2.2 Поддерживаемые конфигурации и микросхмы

Таблица 25.2.

№	F	M	L	K	CF	CS	HD	S	N	Ntotal	Микросхема
1	1	1	2	32	0	0	1	1	16	16	ADC16DX370
2	1	2	4	32	0	0	1	1	16	16	ADS42JB46, ADS42JB49, ADS42JB69
3	2	1	1	32	0	0	0	1	16	16	ADC14X250
4	2	2	2	32	0	0	0	1	16	16	ADC32J42, ADC32J43, ADC32J44, ADC32J45
5	2	2	2	32	0	1	0	1	15	16	
6	2	2	4	32	0	1	0	2	15	16	ADC12J1600, ADC12J2700, ADC12J4000
7	2	2	4	32	0	0	1	2	16	16	ADS54J20
8	2	2	2	8	0	0	0	1	16	16	
9	2	4	4	32	0	0	0	1	16	16	ADC34J22, ADC34J23, ADC34J24, ADC34J25, ADC32RF42
10	4	2	1	32	0	1	0	1	15	16	
11	4	2	1	32	0	0	0	1	16	16	ADC32J22, ADC32J23, ADC32J24, ADC32J25

№	F	M	L	K	CF	CS	HD	S	N	Ntotal	Микросхема
12	4	2	2	32	0	0	0	2	16	16	ADS54J20
13	4	4	2	32	0	0	0	1	16	16	ADC34J42, ADC34J43, ADC34J44, ADC34J45
14	4	2	4	16	0	0	0	4	16	16	ADS54J20
15	4	4	4	32	0	0	0	2	16	16	
16	8	4	1	32	0	1	0	1	15	16	ADC12DJ2700, ADC12DJ3200
17	8	8	4	32	1	0	0	1	16	16	

25.1.3 Структурная схема

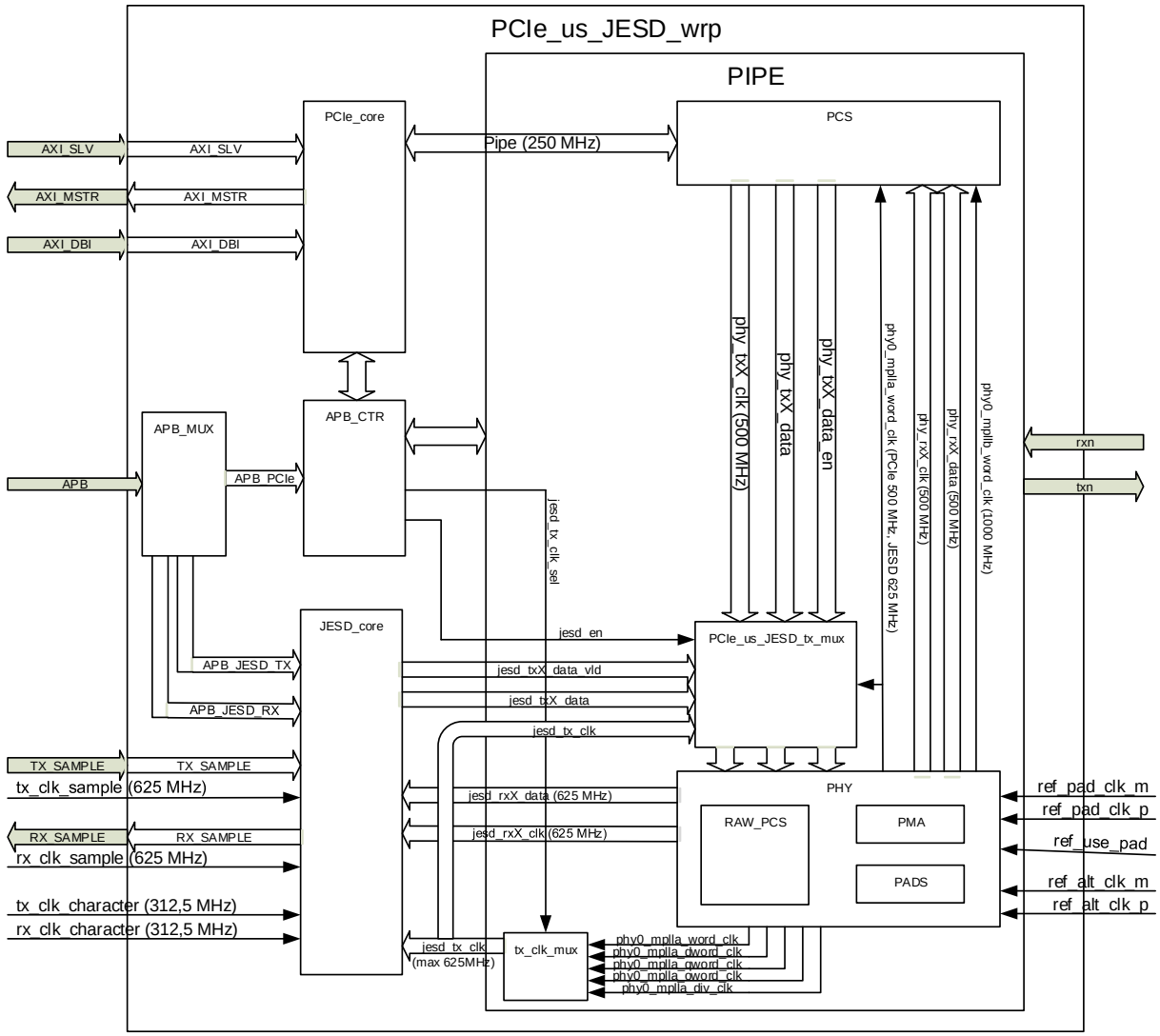


Рисунок 25.3. Схема подключения контроллеров PCIe и Jesd204b

Структурная схема включает в себя следующие основные блоки:

- Контроллер PCI Express
- Контроллер Jesd204b
- Блок регистров APB_CTR
- Блок PIPE, включающий в себя PCS и PHY

25.2 Регистры

Следующие блоки содержат программно доступные регистры:

- Блок APB_CTR
- Контроллер PCI Express
- Блок PHY
- Блок JESD_TX
- Блок JESD_RX

Доступ к регистрам блоков APB_CTR, JESD_TX и JESD_RX осуществляется по APB интерфейсу. Доступ к регистрам PCI Express и PHY осуществляется по интерфейсу AXI (PCI_BDI).

Базовые адреса блоков APB_CTR, JESD_TX, JESD_RX и PCI_DBI представлены в Разделе 6.2.

Смещение адреса регистров PCI Express и PHY относительно базового адреса PCI_DBI представлено в Таблица 25.3.

Таблица 25.3. Адресное пространство доступа к регистрам по AXI интерфейсу

Смещение	Конечный адрес	Описание
0x0	0x3C	TYPE0_HDR
0x40	0x44	PM_CAP
0x50	0x64	MSI_CAP
0x70	0xA0	PCIE_CAP
0xB0	0xB8	MSIX_CAP
0x100	0x144	AER_CAP
0x148	0x184	VC_CAP
0x198	0x1A8	SPCIE_CAP
0x700	0xB74	PORT_LOGIC
0x300000	0x303F20	ATU_CAP (32 входящих и 32 исходящих региона)
0x380000	0x380920	DMA_CAP (4 канала записи, 4 канала чтения)

25.2.1 Регистры блока APB_CTR

Перечень регистров приведен в Таблица 25.4.

Таблица 25.4. Перечень программно-доступных регистров блока APB_CTR

Смещение	Условное Обозначение	Название регистра	Тип доступа
0	SYS_CTRL	System control register	RW

Смещение	Условное Обозначение	Название регистра	Тип доступа
0x0004	INT_ST	Interrupt status	RW
0x0008	AER_INT	AER Interrupt status	RW
0x000C	CAP_INT	CAP Interrupt status	RW
0x0010	EDMA_INTx_INT	DMA Interrupt	RW
0x0014	RADM_CPL_TIMEOUT	RADM CPL timeout attribute	RW
0x0018	TRGT_CPL_TIMEOUT	TRGT CPL timeout attribute	RW
0x001C	PM_CSR	Power management status	RO
0x0020	PM_UNLOCK_ERR	Power management, unlock and error control register	RW
0x0024	GMR_ST	General message reception (GMR) control register	RW
0x0028	GMR_MSG1	1st GMR payload register	RO
0x002C	GMR_MSG2	2nd GMR payload register	RO
0x0030	GMR_MSG_REQ_ID	MSG requester ID	RO
0x0034	DEBUG_ST	Debug status register	RW
0x0038	RFC_DATA0	RFC data low	RO
0x003C	RFC_DATA1	RFC data hi	RO
0x0040	CXPL_DEBUG_INFO1	Debug info low	RO
0x0044	CXPL_DEBUG_INFO2	Debug info hi	RO
0x0048	MSI_ATTR	MSI message attribute	RW
0x004C	MSI_REQ	MSI request register	RW
0x0050	MSI_PENDING	MSI pending	RW
0x0054	MSI_MASK	MSI mask	RO
0x0058	MSI_ADDR0	MSI address low	RO
0x005C	MSI_ADDR1	MSI address hi	RO
0x0060	MSI_DATA	MSI data	RO
0x0064	MSI_CTRL_IO	MSI IO	RO

Смещение	Условное Обозначение	Название регистра	Тип доступа
0x0068	MSI_CTRL_INT	MSI interrupt status	RO
0x006C	VMI_ATTR1	VMI message attribute 1st register	RW
0x0070	VMI_ATTR2	VMI message attribute 2nd register	RW
0x0074	VMI_DATA1	VMI data 1	RW
0x0078	VMI_DATA2	VMI data 2	RW
0x007C	VMI_REQ	VMI request register	WO
0x0080	CFG_ERR	Error status register	RO
0x0100	MSTR_RMISC	AXI Master read misc information	RW
0x0104	MSTR_AWMISC0	AXI Master write misc information 1st word	RO
0x0108	MSTR_AWMISC1	AXI Master write misc information 2nd word	RO
0x010C	MSTR_AWMISC_HDR0	AXI Master 3rd header DW	RO
0x0110	MSTR_AWMISC_HDR1	AXI Master 4th header DW	RO
0x0114	MSTR_ARMISC0	AXI Master read transaction associated misc information 1st word	RO
0x0118	MSTR_ARMISC1	AXI Master read transaction associated misc information 2nd word	RO
0x011C	MSTR_AMISC_LAST	AXI Master last TLP bit	RO
0x0120	MSTR_AMISC_DMA	AXI Master Transaction’s DMA bit’s	RO
0x0200	SLV_AWMISC	AXI Slave write misc information	RW
0x0204	SLV_AWMISC_HDR0	AXI Slave 3rd header DW	RW
0x0208	SLV_AWMISC_HDR1	AXI Slave 4th header DW	RW

Смещение	Условное Обозначение	Название регистра	Тип доступа
0x020C	SLV_ARMISC	AXI Slave read transaction associated misc information	RW
0x0210	SLV_AMISC_ATU	AXI Slave write and read request internal ATU bypass	RW
0x0214	SLV_MISC	AXI Slave read and write misc information	RO
0x0218	SLV_WMISC	AXI Slave write data misc information	RW
0x0300	sys_jesd_en	Выбор режима работы PCIe или JESD	RW
0x0304	jesd_csr		RW
0x0308	sram_crt	Регистр управления доступа к памяти sram phy контроллера	RW
0x030C	protocol0_ext_bs	Регистр установки параметров boundary scan	RW
0x0310	protocol0_ext_bandwidth	Bandwidth control for MPLLA/B	RW
0x0314	protocol0_ext_mplla_div		RW
0x0318	protocol0_ext_mplla_ssc		RW
0x031C	protocol0_ext_mpllb_div		RW
0x0320	protocol0_ext_mpllb_ssc		RW
0x0324	protocol0_ext_ref_and_rx_adapt		RW
0x0328	protocol0_ext_rx_adapt_mode		RW
0x032C	protocol0_ext_rx_cdr_ppm_max_g1		RW
0x0330	protocol0_ext_rx_cdr_ppm_max_g2		RW
0x0334	protocol0_ext_rx_cdr_ppm_max_g3		RW
0x0338	protocol0_ext_rx_cdr_ppm_max_g4		RW
0x033C	protocol0_ext_rx_cdr_vco_freqband		RW
0x0340	protocol0_ext_rx_cdr_vco		RW
0x0344	protocol0_ext_rx_delta_iq_g12		RW
0x0348	protocol0_ext_rx_delta_iq_g34		RW

Смещение	Условное Обозначение	Название регистра	Тип доступа
0x034C	protocol0_ext_rx_dfe_bypass_and_att_lvl_g12		RW
0x0350	protocol0_ext_rx_dfe_bypass_and_att_lvl_g34		RW
0x0354	protocol0_ext_rx_eq_ctle_boost_g1		RW
0x0358	protocol0_ext_rx_eq_ctle_boost_g2		RW
0x035C	protocol0_ext_rx_eq_ctle_boost_g3		RW
0x0360	protocol0_ext_rx_eq_ctle_boost_g4		RW
0x0364	protocol0_ext_rx_eq_ctle_pole		RW
0x0368	protocol0_ext_rx_eq_dfe_tap1_g1		RW
0x036C	protocol0_ext_rx_eq_dfe_tap1_g2		RW
0x0370	protocol0_ext_rx_eq_dfe_tap1_g3		RW
0x0374	protocol0_ext_rx_eq_dfe_tap1_g4		RW
0x0378	protocol0_ext_rx_eq_vga1_gain_g12		RW
0x037C	protocol0_ext_rx_eq_vga1_gain_g34		RW
0x0380	protocol0_ext_rx_eq_vga2_gain_g12		RW
0x0384	protocol0_ext_rx_eq_vga2_gain_g34		RW
0x0388	protocol0_ext_rx_los		RW
0x038C	protocol0_ext_rx_misc_g1		RW
0x0390	protocol0_ext_rx_misc_g2		RW
0x0394	protocol0_ext_rx_misc_g3		RW
0x0398	protocol0_ext_rx_misc_g4		RW
0x039C	protocol0_ext_rx_ref_ld_val		RW
0x03A0	protocol0_ext_rx_vco_ld_val_g12		RW
0x03A4	protocol0_ext_rx_vco_ld_val_g34		RW
0x03A8	protocol0_ext_sup_misc		RW
0x03AC	protocol0_ext_tx_eq_main_g1		RW
0x03B0	protocol0_ext_tx_eq_main_g2		RW
0x03B4	protocol0_ext_tx_eq_main_g3		RW
0x03B8	protocol0_ext_tx_eq_main_g4		RW
0x03BC	protocol0_ext_tx_eq_ovrd		RW
0x03C0	protocol0_ext_tx_eq_post_g1		RW

Смещение	Условное Обозначение	Название регистра	Тип доступа
0x03C4	protocol0_ext_tx_eq_post_g2		RW
0x03C8	protocol0_ext_tx_eq_post_g3		RW
0x03CC	protocol0_ext_tx_eq_post_g4		RW
0x03D0	protocol0_ext_tx_eq_pre_g1		RW
0x03D4	protocol0_ext_tx_eq_pre_g2		RW
0x03D8	protocol0_ext_tx_eq_pre_g3		RW
0x03DC	protocol0_ext_tx_eq_pre_g4		RW
0x03E0	protocol0_ext_tx_misc_g1		RW
0x03E4	protocol0_ext_tx_misc_g2		RW
0x03E8	protocol0_ext_tx_misc_g3		RW
0x03EC	protocol0_ext_tx_misc_g4		RW
0x03F0	protocol0_ext_tx_vboost_lvl		RW
0x03F4	protocol0_ext_sup_misc_recal_bank		RW
0x03F8	rst_status	Регистр статуса сигналов сброса	RO

25.2.1.1 Регистр SYS_CTRL

Формат регистра SYS_CTRL приведен в следующей таблице:

Таблица 25.5. Формат регистра SYS_CTRL

Разряды	Название	Описание		Сброс
31	override_app_ltssm_en	Разрешение перезаписи входного значения с порта app_ltssm_en_in на значение бита app_ltssm_en	RW	0
30	dbi_ro_wr_disable	0 – MISC_CONTROL_1_OFF.DBI_RO_WR_EN поле регистра равно 1 и регистры доступны на запись и чтение 1 – MISC_CONTROL_1_OFF.DBI_RO_WR_EN поле регистра равно 0 и регистры доступны только на чтение	RW	0
29	pf_req_retry_en	Дает возможность откладывать входящие конфигурационные запросы до окончания инициализации PHY	RW	0
28	holp_phy_reset	Удержание PHY под сбросом во время конфигурации регистров во время последовательности сброса по напряжению питания	RW	0
27	cfg_eml_control	Electromechanical Interlock Control	RO	0

Разряды	Название	Описание		Сброс
26	sys_eml_interlock_engaged	System Electromechanical Interlock Engaged	RW	0
25	sys_cmd_cpled_int	Command completed interrupt. Означает, что “Hot-Plug” контроллер выполнил команду	RW	0
24	sys_pre_det_chged	Presence Detect Changed. Означает, что детектор присутствия карты изменил свое состояние	RW	0
23	sys_mrl_sensor_chged	MRL Sensor Changed. Означает, что состояние MRL сенсора изменилось	RW	0
22	sys_pwr_fault_det	Power Fault Detected. Определяет, что контроль напряжения установил отсутствие питания в слоте	RW	0
21	sys_mrl_sensor_state	MRL Sensor State 0 – MRL закрыт 1 – MRL открыт	RW	0
20	sys_pre_det_state	Presence Detect Slot. 0 – Слот пуст 1 – Карта установлена в слот	RW	0
19	sys_atten_button_pressed	Attention Button Pressed	RW	0
18	sys_aux_pwr_det	Auxiliary Power Detected. Используется для информирования о присутствии вспомогательного напряжения (Vaux)	RW	0
17	apps_pm_xmt_pme	Wake up. Запись единицы формирует однотактовый импульс, для выхода из состояния D1, D2 или D3. Во время “Wake-up” контроллер отправляет PM_PME сообщение. Идентичен биту outband_pwrup_cmd	WO	0
16	outband_pwrup_cmd	Wake up. Запись единицы формирует однотактовый импульс, для выхода из состояния D1, D2 или D3. Во время “Wake-up” контроллер отправляет PM_PME сообщение. Идентичен биту apps_pm_xmt_pme	WO	0
15	-	Резерв	RO	0
14	app_xfer_pending	Установка бита предотвращает переход в состояние L1 и означает, что ожидается транзакция.	RW	0
13	app_req_exit_l1	Запрос на выход из состояния L1 ASPM	RW	0
12	app_req_entr_l1	Запрос на переход в состояние L1 ASPM	RW	0
11	tx_lane_flip_en	Разрешение реверса линий передачи	RW	0
10	rx_lane_flip_en	Разрешение реверса линий приема	RW	0
9:8	-	Резерв	RO	0
7	app_clk_pm_en	Разрешение “Clock PM”	RW	0

Разряды	Название	Описание		Сброс
6	app_init_rst	Запись единицы отправляет запрос hot reset на “upstream” порт	WO	0
5	app_req_retry_en	Установка бита позволяет отсрочить конфигурационные запросы до конца инициализации	RW	0
4	app_ltssm_en	Разрешение запуска инициализации (link training) Значении этого бита транслируется на вход контроллера при override_app_ltssm_en=1, иначе на вход контроллера берется сигнал с внешнего вывода	RW	0
3:0	device_type	Определяет тип порта: 4’b0000 – PCI Express endpoint 4’b0001 – Legacy PCI Express endpoint 4’b0100 – Root port Другие кодировки не поддерживаются Значение данного поля транслируется в регистр “PCI Express Capabilities Register”	RW	0

25.2.1.2 Регистр INT_ST

Регистр системных прерываний INT_ST. Значения полей hp_pme, hp_int, hp_msi, cfg_link_auto_bw_int, cfg_link_auto_bw_msi, cfg_bw_mgt_int, cfg_bw_mgt_msi и cfg_link_eq_req_int объединены по «ИЛИ» в один сигнал system_interrupt, который выдается на верхний уровень как запрос на прерывание

Формат регистра INT_ST приведен в следующей таблице:

Таблица 25.6. Формат регистра INT_ST

Разряды	Название	Описание		Сброс
31	sys_int	При переключении из 0 в 1 контроллер передает сообщение “Assert_INTA”. При переключении из 1 в 0 контроллер передает сообщение “Deassert_INTA”.	RW	0
30:21	-	Резерв	RO	0
20	radm_vendor_msg	Дублирует значение поля radm_vendor_msg регистра GMR_ST	RO	0
19	cfg_link_eq_req_int	Означает, что установлены биты “Link Equalization Request” регистра “Link Status 2” и “ Link Equalization Request Interrupt Enable” регистра “Link Control 3”	RO	0
18	cfg_bw_mgt_msi	Контроллер устанавливает данный бит если все условия выполняются: MSI или MSI-X разрешены “Link Bandwidth Management Status” 14-й бит регистра “Link Control Status” изменяется “Link Bandwidth Management Interrupt Enable” 10-й бит в регистре “ Link Control” установлен в 1 Сброс записью единицы в данный разряд	RW	0

Разряды	Название	Описание		Сброс
17	cfg_bw_mgt_int	Контроллер устанавливает данный бит если все условия выполняются: INTx Assertion Disable бит в “Command” регистре установлен в 0 “Bandwidth Management Interrupt Enable” бит в регистре “Link Control” установлен в 1 “Bandwidth Management Interrupt Status” бит регистра “Link Status” установлен в 1	RO	0
16	cfg_link_auto_bw_msi	Контроллер устанавливает данный бит если все условия выполняются: MSI или MSI-X разрешены “Link Autonomous Bandwidth Status” 15-й бит регистра “Link Status” изменяется “Link Autonomous Bandwidth Interrupt Enable” 11-й бит в регистре “ Link Control” установлен в 1	RW	0
15	cfg_link_auto_bw_int	Контроллер устанавливает данный бит если все условия выполняются: INTx Assertion Disable бит в “Command” регистре установлен в 0 “Link Autonomous Bandwidth Interrupt Enable” бит в регистре “Control” установлен в 1 “Link Autonomous Bandwidth Interrupt Status” бит регистра “Link Status” установлен в 1	RO	0
14	hp_msi	Контроллер устанавливает данный бит если все условия выполняются: MSI или MSI-X разрешены “Hot-Plug” прерывания разрешены в регистре “Slot Control” Любой из битов регистра “Slot Status” переключился из 0 в 1 и соответствующее событие разрешено в регистре “Slot Control” Сброс записью единицы в данный разряд	RW	0
13	hp_int	Контроллер устанавливает данный бит если оба условия выполняются: INTx Assertion Disable бит в “Command” регистре установлен в 0 Любой из битов регистра “Slot Status” установлен в 1 и соответствующее событие разрешено в регистре “Slot Control”	RO	0
12	hp_pme	Контроллер устанавливает данный бит если оба условия выполняются: PME Enable бит в “Power Management Control and Status” регистре установлен в 1 Любой из битов регистра “Slot Status” переключился из 0 в 1 и соответствующее событие разрешено в регистре “Slot Control”	RO	0
11:9	-	Резерв	RO	0
8	cfg_int_disable	0 - отправка сообщений INTx разрешена 1 - отправка сообщений INTx запрещена	RO	0

Разряды	Название	Описание		Сброс
7:0	cfg_int_pin	Отображает значение поля “BRIDGE_CTRL_PIN_INT_LINE” регистра “Interrupt Pin”	RO	0

25.2.1.3 Регистр AER_INT

Регистр прерываний AER_INT. Значения полей cfg_aer_rc_err_int и cfg_aer_rc_err_msi объединены по «ИЛИ» в один сигнал error_interrupt, который выдается на верхний уровень как запрос на прерывание

Формат регистра AER_INT приведен в следующей таблице:

Таблица 25.7. Формат регистра AER_INT

Разряды	Название	Описание		Сброс
31:10	-	Резерв	RO	0
9	cfg_aer_rc_err_msi	Контроллер устанавливает данный бит если все условия выполняются: MSI или MSI-X разрешены Установлен бит ошибки в “Root Error Status” регистре в 1 Установлен бит разрешающий отчет о данной ошибке в регистре “Root Error Command” Сброс записью единицы в данный разряд	RW	0
8	cfg_aer_rc_err_int	Контроллер устанавливает данный бит если оба условия выполняются: Установлен бит ошибки в “Root Error Status” регистре в 1 Установлен бит разрешающий отчет о данной ошибке в регистре “Root Error Command”	RO	0
7:5	-	Резерв	RO	0
4:0	cfg_aer_int_msg_num	Отображает биты [31:27] регистра “Root Error Status”	RO	0

25.2.1.4 Регистр CAP_INT

Регистр прерываний CAP_INT. Значения полей cfg_pme_int и cfg_pme_msi объединены по «ИЛИ» в один сигнал pme_interrupt, который выдается на верхний уровень как запрос на прерывание

Формат регистра CAP_INT приведен в следующей таблице:

Таблица 25.8. Формат регистра CAP_INT

Разряды	Название	Описание		Сброс
31:10	-	Резерв	RO	0
9	cfg_pme_msi	Контроллер устанавливает данный бит если все условия выполняются:	RW	0

Разряды	Название	Описание		Сброс
		MSI или MSI-X разрешены “PME Interrupt Enable” бит в “Root Control” регистре установлен в 1 “PME Status” бит регистра “Root Status” установлен в 1 Сброс записью единицы в данный разряд		
8	cfg_pme_int	Контроллер устанавливает данный бит если все условия выполняются: INTx Assertion Disable бит в “Command” регистре установлен в 0 “PME Interrupt Enable” бит в “Root Control” регистре установлен в 1 “PME Status” бит регистра “Root Status” установлен в 1	RO	0
7:5	-	Резерв	RO	0
4:0	cfg_cap_int_msg_num	Отображает биты [13:9] регистра “PCIe Capabilities” Доступен на чтение	RO	0

25.2.1.5 Регистр EDMA_INTx_INT

Регистр прерываний от DMA и Legacy прерываний EDMA_INTx_INT. Значения полей edma_int объединены по «ИЛИ» в один сигнал edma_interrupt, который выдается на верхний уровень как запрос на прерывание. Остальные поля регистра объединены по «ИЛИ» в сигнал intx_interrupt, который выдается на верхний уровень как запрос на прерывание.

Формат регистра EDMA_INTx_INT приведен в следующей таблице:

Таблица 25.9. Формат регистра EDMA_INTx_INT

Разряды	Название	Описание		Сброс
31:24	edma_int	Прерывание от DMA. Устанавливается при завершении работы DMA или при возникновении ошибки	RO	0
23:16	-	Резерв	RO	0
15	deassert_intd_grt	Контроллер отправил сообщение “Deassert_INTD” Сброс записью единицы в данный разряд	RW	0
14	deassert_intc_grt	Контроллер отправил сообщение “Deassert_INTC” Сброс записью единицы в данный разряд	RW	0
13	deassert_intb_grt	Контроллер отправил сообщение “Deassert_INTB” Сброс записью единицы в данный разряд	RW	0
12	deassert_inta_grt	Контроллер отправил сообщение “Deassert_INTA” Сброс записью единицы в данный разряд	RW	0
11	assert_intd_grt	Контроллер отправил сообщение “Assert_INTD” Сброс записью единицы в данный разряд	RW	0
10	assert_intc_grt	Контроллер отправил сообщение “Assert_INTC” Сброс записью единицы в данный разряд	RW	0

Разряды	Название	Описание		Сброс
9	assert_intb_grt	Контроллер отправил сообщение “Assert_INTB” Сброс записью единицы в данный разряд	RW	0
8	assert_inta_grt	Контроллер отправил сообщение “Assert_INTA” Сброс записью единицы в данный разряд	RW	0
7	radm_intd_deasserted	Принято сообщение “Deassert_INTD” Сброс записью единицы в данный разряд	RW	0
6	radm_intc_deasserted	Принято сообщение “Deassert_INTC” Сброс записью единицы в данный разряд	RW	0
5	radm_intb_deasserted	Принято сообщение “Deassert_INTB” Сброс записью единицы в данный разряд	RW	0
4	radm_inta_deasserted	Принято сообщение “Deassert_INTA” Сброс записью единицы в данный разряд	RW	0
3	radm_intd_asserted	Принято сообщение “Assert_INTD” Сброс записью единицы в данный разряд	RW	0
2	radm_intc_asserted	Принято сообщение “Assert_INTC” Сброс записью единицы в данный разряд	RW	0
1	radm_intb_asserted	Принято сообщение “Assert_INTB” Сброс записью единицы в данный разряд	RW	0
0	radm_inta_asserted	Принято сообщение “Assert_INTA” Сброс записью единицы в данный разряд	RW	0

25.2.1.6 Регистр RADM_CPL_TIMEOUT

Регистр таймаута переданных транзакций RADM_CPL_TIMEOUT

Формат регистра RADM_CPL_TIMEOUT приведен в следующей таблице:

Таблица 25.10. Формат регистра RADM_CPL_TIMEOUT

Разряды	Название	Описание		Сброс
31	radm_cpl_timeout	Признак обнаружения таймаут транзакции Сброс записью единицы в данный разряд	RW	0
30:28	radm_timeout_func_num	Номер функции таймаут транзакции	RO	0
27:26	-	Резерв	RO	0
25:24	radm_timeout_cpl_attr	Поле “Attribute” таймаут транзакции	RO	0
23	-	Резерв	RO	0
22:20	radm_timeout_cpl_tc	“Traffic Class” таймаут транзакции	RO	0
19:8	radm_timeout_cpl_len	Длина в байтах таймаут транзакции	RO	0
7:0	radm_timeout_cpl_tag	Поле “tag” таймаут транзакции	RO	0

25.2.1.7 Регистр TRGT_CPL_TIMEOUT

Регистр таймаута ответных транзакций TRGT_CPL_TIMEOUT

Формат регистра TRGT_CPL_TIMEOUT приведен в следующей таблице:

Таблица 25.11. Формат регистра TRGT_CPL_TIMEOUT

Разряды	Название	Описание		Сброс
31	trgt_cpl_timeout	Признак обнаружения таймаут ответной транзакции Сброс записью единицы в данный разряд	RW	0
30:23	-	Резерв	RO	0
22:20	trgt_timeout_func_num	Номер функции таймаут ответной транзакции	RO	0
19:18	-	Резерв	RO	0
17:16	trgt_timeout_cpl_attr	Поле “Attribute” таймаут ответной транзакции	RO	0
15	-	Резерв	RO	0
14:12	trgt_timeout_cpl_tc	“Traffic Class” таймаут ответной транзакции	RO	0
11:0	trgt_timeout_cpl_len	Длина в байтах таймаут ответной транзакции	RO	0

25.2.1.8 Регистр PM_CSR

Формат регистра PM_CSR приведен в следующей таблице:

Таблица 25.12. Формат регистра PM_CSR

Разряды	Название	Описание		Сброс
31:29	-	Резерв	RO	0
28:24	pm_slave_state	Power management slave FSM state	RO	0
23:21	-	Резерв	RO	0
20:16	pm_master_state	Power management master FSM state	RO	0
15	wake	Запрос Wake up	RO	0
14	pm_l1_entry_started	Выполнение перехода в состояние L1	RO	0
13	pm_en_core_clk	Признак гетирования частоты core_clk	RO	0
12:10	pm_curnt_dstate	Отображает текущее состояние питания Для отладочных целей	RO	0
9:7	pm_dstate	Текущее состояние машины управления питанием: 000b – D0 001b – D1 010b – D2 011b – D3 100b – Uninitilized	RO	0
6	pm_pme_en	“PME Enable” бит регистра “PMCSR”	RO	0

Разряды	Название	Описание		Сброс
5	pm_linkst_in_l0s	Управление питанием в состоянии L0s	RO	0
4	pm_linkst_in_l1	Управление питанием в состоянии L1	RO	0
3	pm_linkst_in_l2	Управление питанием в состоянии L2	RO	0
2	pm_linkst_l2_exit	Управление питанием осуществляет выход из состояния L2	RO	0
1	pm_status	“PME Status” бит регистра “PMCSR”	RO	0
0	aux_pm_en	“Auxiliary Power Enable” бит регистра “Device Control”	RO	0

25.2.1.9 Регистр PM_UNLOCK_ERR

Регистр Power management, unlock and error control register PM_UNLOCK_ERR

Формат регистра PM_UNLOCK_ERR приведен в следующей таблице:

Таблица 25.13. Формат регистра PM_UNLOCK_ERR

Разряды	Название	Описание		Сброс
31	apps_pm_xmt_turnoff	Запрос на передачу “PM_Turn_off” сообщения	WO	0
30	app_unlock_msg	Запрос на передачу “Unlock” сообщения	WO	0
29:8	-	Резерв	RO	0
7	cfg_sys_err_rc	Обнаружена ошибка. Выставляется при ERR_COR, ERR_FATAL или ERR_NONFATAL при условии, что соответствующие разрешающие биты регистра “Root Control” установлены Сброс записью единицы в данный разряд	RW	0
6	radm_msg_unlock	Контроллер принял сообщение “Unlock” Сброс записью единицы в данный разряд	RW	0
5	radm_correctable_err	Контроллер принял сообщение “ERR_COR” Сброс записью единицы в данный разряд	RW	0
4	radm_nonfatal_err	Контроллер принял сообщение “ERR_NONFATAL” Сброс записью единицы в данный разряд	RW	0
3	radm_fatal_err	Контроллер принял сообщение “ERR_FATAL” Сброс записью единицы в данный разряд	RW	0
2	radm_pm_pme	Контроллер принял сообщение “PM_PME” Сброс записью единицы в данный разряд	RW	0
1	radm_pm_to_ack	Контроллер принял сообщение “PME_TO_Ack” Сброс записью единицы в данный разряд	RW	0
0	radm_pm_turnoff	Контроллер принял сообщение “Turnoff” Сброс записью единицы в данный разряд	RW	0

25.2.1.10 Регистр GMR_ST

Формат регистра GMR_ST приведен в следующей таблице:

Таблица 25.14. Формат регистра GMR_ST

Разряды	Название	Описание		Сброс
31	radm_vendor_msg	Контроллер принял “vendor” сообщение	RW	0
30:5	-	Резерв	RO	0
4	cfg_send_cor_err	Отправлено “Correctable Error” в Root Complex	RO	0
3	cfg_send_nf_err	Отправлено “Non-Fatal Error” в Root Complex	RO	0
2	cfg_send_f_err	Отправлено “Fatal Error” в Root Complex	RO	0
1	surprise_down_err	Означает, что было событие “surprise down”	RO	0
0	cfg_br_ctrl_serren	Значение “PF’s SERR# Enable” из регистра “Bridge Control Register”	RO	0

25.2.1.11 Регистр GMR_MSG1

Формат регистра GMR_MSG1 приведен в следующей таблице:

Таблица 25.15. Формат регистра GMR_MSG1

Разряды	Название	Описание		Сброс
31:0	radm_msg_payload	Байты с 12 по 15. 4-е слово заголовка “vendor” сообщения. [7:0] – 15-й байт	RO	0

25.2.1.12 Регистр GMR_MSG2

Формат регистра GMR_MSG2 приведен в следующей таблице:

Таблица 25.16. Формат регистра GMR_MSG2

Разряды	Название	Описание		Сброс
31:0	radm_msg_payload	Байты с 8 по 11. 3-е слово заголовка “vendor” сообщения	RO	0

25.2.1.13 Регистр GMR_MSG_REQ_ID

Формат регистра GMR_MSG_REQ_ID приведен в следующей таблице:

Таблица 25.17. Формат регистра GMR_MSG_REQ_ID

Разряды	Название	Описание		Сброс
31:16	-	Резерв	RO	0
15:0	radm_msg_req_id	ID запроса принятого сообщения [15:8] – Bus number	RO	0

Разряды	Название	Описание		Сброс
		[7:3] – Device number [2:0] – Function number		

25.2.1.14 Регистр DEBUG_ST

Формат регистра DEBUG_ST приведен в следующей таблице:

Таблица 25.18. Формат регистра DEBUG_ST

Разряды	Название	Описание		Сброс
31	smlh_link_up	PHY Link up/down indicator	RO	0
30:20	-	Резерв	RO	0
19:16	radm_q_not_empty	Очередь приема содержит TLP данные/заголовок Сброс записью единицы в данный разряд	RW	0
15:12	radm_qoverflow	Означает, что одна или несколько P/NP/CPL очередь переполнены Сброс записью единицы в данный разряд	RW	0
11:10	-	Резерв	RO	0
9:8	rtlh_rfc_upd	Контроллер принял “update DLLP”01 – был принят один “update DLLP”11 – были приняты два “update DLLP” одновременно Сброс записью единицы в данный разряд	RW	0
7:5	-	Резерв	RO	0
4	radm_xfer_pending	Контроллер ожидает ответа от AXI	RO	0
3	edma_xfer_pending	DMA канал чтения или записи не закончил передачу данных	RO	0
2	brdg_dbi_xfer_pending	DBI AXI слэйв ожидает ответа от контроллера	RO	0
1	brdg_slv_xfer_pending	AXI слэйв ожидает ответа от контроллера	RO	0
0	rdlh_link_up	Data link layer up/down indicator Означает, что data link Layer готов принимать и передавать пакеты	RO	0

25.2.1.15 Регистр RFC_DATA0

Формат регистра RFC_DATA0 приведен в следующей таблице:

Таблица 25.19. Формат регистра RFC_DATA0

Разряды	Название	Описание		Сброс
31:0	rtlh_rfc_data	Данные приняты в “flow control update DLLP” Валидны при rtlh_rfc_upd=01 или rtlh_rfc_upd=11	RO	0

25.2.1.16 Регистр RFC_DATA1

Формат регистра RFC_DATA1 приведен в следующей таблице:

Таблица 25.20. Формат регистра RFC_DATA1

Разряды	Название	Описание		Сброс
31:0	rtlh_rfc_data	Данные приняты во втором “flow control update DLLP” rtlh_rfc_upd=11	RO	0

25.2.1.17 Регистр CXPL_DEBUG_INFO1

Формат регистра CXPL_DEBUG_INFO1 приведен в следующей таблице:

Таблица 25.21. Формат регистра CXPL_DEBUG_INFO1

Разряды	Название	Описание		Сброс
31:0	cxpl_debug_info	[31:28]: smlh_ts_link_ctrl: Link control bits advertised by link partner (Reserved for M-PCIe) [27]: smlh_ts_lane_num_is_k237: Currently receiving k237 (PAD) in place of lane number (Reserved for M-PCIe) [26]: smlh_ts_link_num_is_k237: Currently receiving k237 (PAD) in place of link number (Reserved for M-PCIe) [25]: rmlh_rcvd_idle[0]: Receiver is receiving logical idle [24]: rmlh_rcvd_idle[1]: 2n symbol is also idle (16bit PHY interface only) [23:8]: mac_phy_txdata: PIPE(RMMI for M-PCIe) Transmit data [7:6]: mac_phy_txdatak: PIPE(RMMI for M-PCIe) transmit K indication [5:0]: smlh_ltssm_state: LTSSM current state	RO	0

25.2.1.18 Регистр CXPL_DEBUG_INFO2

Формат регистра CXPL_DEBUG_INFO2 приведен в следующей таблице:

Таблица 25.22. Формат регистра CXPL_DEBUG_INFO2

Разряды	Название	Описание		Сброс
31:0	cxpl_debug_info	[31]: smlh_scrambler_disable: Scrambling disabled for the link [30]: smlh_link_disable: TSSM in DISABLE state. Link inoperable [29]: smlh_link_in_training: LTSSM performing link training [28]: smlh_ltssm_in_pollconfig: LTSSM is in Polling.Configuration state [27]: smlh_training_rst_n: LTSSM-negotiated link reset [26:23]: 0000b: Reserved [22]: mac_phy_txdetectrx_loopback: PIPE receiver detect/loopback request (Reserved for M-PCIe) [21]: mac_phy_txelecidle[0]: PIPE transmit electrical idle request (RMMI TX_Burst[0] (TX_Burst initiates a BURST) for M-PCIe) [20]: mac_phy_txcompliance[0]: PIPE transmit compliance request (RX Electrical Idle at internal PIPE I/F(from Rpa to Smlh module) for M-PCIe) [19]: app_init_rst: Application request to initiate training reset [18:16]: 000b: Reserved [15:8]: rmlh_ts_link_num: Link number advertised/confirmed by link partner (Reserved for M-PCIe) [7:6]: 00b: Reserved [5]: xmtbyte_skip_sent: A skip ordered set has been transmitted	RO	0

Разряды	Название	Описание		Сброс
		[4]: smlh_link_up: LTSSM reports PHY link up or LTSSM is in Loopback.Active for Loopback Master [3]: smlh_inskip_rcv: Receiver reports skip reception (Reserved for M-PCIe) [2]: smlh_ts1_rcvd: TS1 training sequence received (pulse) (Reserved for M-PCIe) [1]: smlh_ts2_rcvd: TS2 training sequence received (pulse) (Reserved for M-PCIe) [0]: smlh_rcvd_lane_rev: Receiver detected lane reversal (Reserved for M-PCIe)		

25.2.1.19 Регистр MSI_ATTR

Формат регистра MSI_ATTR приведен в следующей таблице:

Таблица 25.23. Формат регистра MSI_ATTR

Разряды	Название	Описание		Сброс
31	cfg_msi_en	Признак разрешения MSI сообщений	RO	0
30	cfg_msi_64	Бит “64-bit Address Capable” регистра “MSI Control”	RO	0
29:27	cfg_multi_msi_en	Поле “Multiple Message Enable” регистра “MSI Control”	RO	0
26	cfg_msi_ext_data_en	Признак разрешения расширенных данных в MSI сообщении	RO	0
25:11	-	Резерв	RO	0
10:8	ven_msi_func_num	Номер функции MSI запроса Запись в это поле не доступна, пока установлен бит msi_request регистра MSI_REQ	RW	0
7:5	ven_msi_tc	“Traffic Class” MSI запроса Запись в это поле не доступна, пока установлен бит msi_request регистра MSI_REQ	RW	0
4:0	ven_msi_vector	Используется для изменения пяти младших бит регистра MSI Data Запись в это поле не доступна, пока установлен бит msi_request регистра MSI_REQ	RW	0

25.2.1.20 Регистр MSI_REQ

Формат регистра MSI_REQ приведен в следующей таблице:

Таблица 25.24. Формат регистра MSI_REQ

Разряды	Название	Описание		Сброс
31:1	-	Резерв	RO	0
0	msi_request	Запись в это поле 1 инициирует запрос контроллеру на передачу MSI сообщения При подтверждении контроллером задания это поле аппаратно сбрасывается в 0	WO	0

25.2.1.21 Регистр MSI_PENDING

Формат регистра MSI_PENDING приведен в следующей таблице:

Таблица 25.25. Формат регистра MSI_PENDING

Разряды	Название	Описание		Сброс
31:0	cfg_msi_pending	Указывает какая функция имеет не завершенные сообщения. Используется для определения значения регистра “Vector Interrupt Pending”	RW	0

25.2.1.22 Регистр MSI_MASK

Регистр маски MSI сообщений

Формат регистра MSI_MASK приведен в следующей таблице:

Таблица 25.26. Формат регистра MSI_MASK

Разряды	Название	Описание		Сброс
31:0	cfg_msi_mask	Содержит значение регистра “Per Vector Mask”	RO	0

25.2.1.23 Регистр MSI_ADDR0

Регистр адреса MSI сообщений

Формат регистра MSI_ADDR0 приведен в следующей таблице:

Таблица 25.27. Формат регистра MSI_ADDR0

Разряды	Название	Описание		Сброс
31:0	cfg_msi_addr	Младшие 32 бита регистра “MSI Address”	RO	0

25.2.1.24 Регистр MSI_ADDR1

Регистр адреса MSI сообщений

Формат регистра MSI_ADDR1 приведен в следующей таблице:

Таблица 25.28. Формат регистра MSI_ADDR1

Разряды	Название	Описание		Сброс
31:0	cfg_msi_addr	Старшие 32 бита регистра “MSI Address”	RO	0

25.2.1.25 Регистр MSI_DATA

Регистр данных MSI сообщения

Формат регистра MSI_DATA приведен в следующей таблице:

Таблица 25.29. Формат регистра MSI_DATA

Разряды	Название	Описание		Сброс
31:0	cfg_msi_data	Содержит значение регистра “MSI Data”	RO	0

25.2.1.26 Регистр MSI_CTRL_IO

Формат регистра MSI_CTRL_IO приведен в следующей таблице:

Таблица 25.30. Формат регистра MSI_CTRL_IO

Разряды	Название	Описание		Сброс
31:0	cfg_msi_ctrl_io	GPIO регистр прерываний MSI	RO	0

25.2.1.27 Регистр MSI_CTRL_INT

Регистр признака приема MSI сообщения MSI_CTRL_INT. Значение поля msi_ctrl_int преобразуется в сигнал msi_interrupt и передается на верхний уровень как запрос на прерывание

Формат регистра MSI_CTRL_INT приведен в следующей таблице:

Таблица 25.31. Формат регистра MSI_CTRL_INT

Разряды	Название	Описание		Сброс
31:9	-	Резерв	RO	0
8	msi_ctrl_int	Признак обнаружения DSP портом MSI сообщения	RO	0
7:0	msi_ctrl_int_vec	DSP вектор прерываний MSI	RO	0

25.2.1.28 Регистр VMI_ATTR1

Регистр атрибутов VMI сообщения

Формат регистра VMI_ATTR1 приведен в следующей таблице:

Таблица 25.32. Формат регистра VMI_ATTR1

Разряды	Название	Описание		Сброс
31:16	-	Резерв	RO	0
15:8	ven_msg_tag	Поле “Tag” “vendor” сообщения TLP	RW	0
7:0	ven_msg_code	Код “vendor” сообщения TLP	RW	0

25.2.1.29 Регистр VMI_ATTR2

Регистр атрибутов VMI сообщения

Формат регистра VMI_ATTR2 приведен в следующей таблице:

Таблица 25.33. Формат регистра VMI_ATTR2

Разряды	Название	Описание		Сброс
31:30	-	Резерв	RO	0
29:20	ven_msg_len	Поле “Length” “vendor” сообщения TLP	RW	0
19	-	Резерв	RO	0
18:16	ven_msg_func_num	Поле “Function Number” “vendor” сообщения TLP	RW	0
15:14	-	Резерв	RO	0
13:12	ven_msg_attr	Поле “ATTR” “vendor” сообщения TLP	RW	0
11:10	ven_msg_fmt	Поле “FMT” “vendor” сообщения TLP	RW	0x1
9	ven_msg_ep	Поле “EP” “vendor” сообщения TLP	RW	0
8	ven_msg_td	Поле “TD” “vendor” сообщения TLP	RW	0
7:5	ven_msg_tc	Поле “TC” “vendor” сообщения TLP	RW	0
4:0	ven_msg_type	Поле “Type” “vendor” сообщения TLP	RW	0

25.2.1.30 Регистр VMI_DATA1

Регистр данных VMI сообщения

Формат регистра VMI_DATA1 приведен в следующей таблице:

Таблица 25.34. Формат регистра VMI_DATA1

Разряды	Название	Описание		Сброс
31:0	ven_msg_data	Байты 8-11 (третье слово заголовка) “vendor” сообщения TLP	RW	0

25.2.1.31 Регистр VMI_DATA2

Регистр данных VMI сообщения

Формат регистра VMI_DATA2 приведен в следующей таблице:

Таблица 25.35. Формат регистра VMI_DATA2

Разряды	Название	Описание		Сброс
31:0	ven_msg_data	RW Байты 12-15 (четвертое слово заголовка) “vendor” сообщения TLP	RW	0

25.2.1.32 Регистр VMI_REQ

Регистр запроса на отправку VMI сообщения

Формат регистра VMI_REQ приведен в следующей таблице:

Таблица 25.36. Формат регистра VMI_REQ

Разряды	Название	Описание		Сброс
31:1	-	Резерв	RO	0
0	vmi_request	Запись в это поле 1 инициирует запрос контроллеру на передачу VMI сообщения При подтверждении контроллером задания это поле аппаратно сбрасывается в 0	RW	0

25.2.1.33 Регистр CFG_ERR

Формат регистра CFG_ERR приведен в следующей таблице:

Таблица 25.37. Формат регистра CFG_ERR

Разряды	Название	Описание		Сброс
31:17	-	Резерв	RO	0
16	cfg_uncor_internal_err_sts	Признак обнаружения контроллером "Uncorrectable Internal Error"	RO	0
15	cfg_surprise_down_er_sts	Признак обнаружения контроллером "Surprise Down Error"	RO	0
14	cfg_replay_timer_timeout_err_sts	Признак обнаружения контроллером "Replay Timer Timeout"	RO	0
13	cfg_replay_number_rollover_err_sts	Признак обнаружения контроллером "REPLAY_NUMBER Rollover Error"	RO	0
12	cfg_reg_serren	Значение поля PF's SERR# Enable регистра Command Register of Type0 Header, для установки регистра статуса ошибки внешних VFs	RO	0
11	cfg_rcvr_overflow_err_sts	Признак обнаружения контроллером "Receiver Overflow Error"	RO	0
10	cfg_rcvr_err_sts	Признак обнаружения контроллером "Receiver Error"	RO	0
9	cfg_nf_err_rpt_en	Значение поля PF's Non-Fatal Error Reporting Enable регистра Device Control Register в PCIe Capability, для отправки ERR_MSG во внешние VFs	RO	0
8	cfg_mlf_tlp_err_sts	Признак обнаружения контроллером "Malformed TLP Error"	RO	0
7	cfg_fc_protocol_err_sts	Признак обнаружения контроллером "Flow Control Protocol Error"	RO	0
6	cfg_f_err_rpt_en	Значение поля PF's Fatal Error Reporting Enable регистра in Device Control Register в	RO	0

Разряды	Название	Описание		Сброс
		PCIe Capability, для отправки ERR_MSG во внешние VFs.		
5	cfg_ecrc_err_sts	Признак обнаружения контроллером "ECRC Error"	RO	0
4	cfg_dl_protocol_err_sts	Признак обнаружения контроллером "Data Link Protocol Error"	RO	0
3	cfg_corrected_internal_err_sts	Признак обнаружения контроллером "Corrected Internal Error"	RO	0
2	cfg_cor_err_rpt_en	Значение поля PF's Correctable Error Reporting Enable регистра Device Control Register в PCIe Capability, для отправки ERR_MSG во внешние VFs.	RO	0
1	cfg_bad_tlp_err_sts	Признак обнаружения контроллером "Bad TLP Error"	RO	0
0	cfg_bad_dllp_err_sts	Признак обнаружения контроллером "Bad DLLP Error"	RO	0

25.2.1.34 Регистр MSTR_RMISC

Формат регистра MSTR_RMISC приведен в следующей таблице:

Таблица 25.38. Формат регистра MSTR_RMISC

Разряды	Название	Описание		Сброс
31:8	-	Резерв	RO	0
7	TLP_EP	AXI master read response transaction associated misc information. Не является частью стандартного интерфейса AXI и служит как опционное дополнение к AXI интерфейсу	RW	0
6:0	-	Резерв	RO	0

25.2.1.35 Регистр MSTR_AWMISC0

Формат регистра MSTR_AWMISC0 приведен в следующей таблице:

Таблица 25.39. Формат регистра MSTR_AWMISC0

Разряды	Название	Описание		Сброс
31	mstr_awmisc_info_ep	AXI master write mapping to the EP bit in the PCIe protocol domain	RO	0
30:24	-	Резерв	RO	0

Разряды	Название	Описание		Сброс
23:0	mstr_awmisc_info0	AXI master write misc information. Не является частью стандартного интерфейса AXI и служит как опционное дополнение к AXI интерфейсу. - При типе транзакции MSG, биты 7-0 отображают код сообщения Соответствия битов: [5:0] – Резервные, за исключением MSG TLP. [8:6] – номер BAR из TLP [9] – TLP и I/O [10] – TLP в диапазоне ROM [13:11] – Номер функции TLP [14] – TLP NS бит [15] – TLP RO бит [18:16] – TLP TC бит [23:19] – TLP TYPE	RO	0

25.2.1.36 Регистр MSTR_AWMISC1

Формат регистра MSTR_AWMISC1 приведен в следующей таблице:

Таблица 25.40. Формат регистра MSTR_AWMISC1

Разряды	Название	Описание		Сброс
31:24	-	Резерв	RO	0
23:0	mstr_awmisc_info1	AXI master write misc information. Не является частью стандартного интерфейса AXI и служит как опционное дополнение к AXI интерфейсу. Соответствия битов: [15:0] – TLP requester ID [23:16] – TLP TAG	RO	0

25.2.1.37 Регистр MSTR_AWMISC_HDR0

Формат регистра MSTR_AWMISC_HDR0 приведен в следующей таблице:

Таблица 25.41. Формат регистра MSTR_AWMISC_HDR0

Разряды	Название	Описание		Сброс
31:0	mstr_awmisc_info_hdr_3dw	AXI master write misc information. Не является частью стандартного интерфейса AXI и служит как опционное дополнение к AXI интерфейсу. Содержит третье слово заголовка	RO	0

25.2.1.38 Регистр MSTR_AWMISC_HDR1

Формат регистра MSTR_AWMISC_HDR1 приведен в следующей таблице:

Таблица 25.42. Формат регистра MSTR_AWMISC_HDR1

Разряды	Название	Описание		Сброс
31:0	mstr_awmisc_info_hdr_4dw	AXI master write misc information. Не является частью стандартного интерфейса AXI и служит как опционное дополнение к AXI интерфейсу. Содержит четвертое слово заголовка	RO	0

25.2.1.39 Регистр MSTR_ARMISC0

Формат регистра MSTR_ARMISC0 приведен в следующей таблице:

Таблица 25.43. Формат регистра MSTR_ARMISC0

Разряды	Название	Описание		Сброс
31:24	-	Резерв	RO	0
23:0	mstr_armisc_info0	AXI master read misc information. Не является частью стандартного интерфейса AXI и служит как опционное дополнение к AXI интерфейсу. При типе транзакции MSG, биты 7-0 отображают код сообщения Соответствия битов: [5:0] – Резервные, за исключением MSG TLP. [8:6] – номер BAR из TLP [9] – TLP и I/O [10] – TLP в диапазоне ROM [13:11] – Номер функции TLP [14] – TLP NS бит [15] – TLP RO бит [18:16] – TLP TC бит [23:19] – TLP TYPE	RO	0

25.2.1.40 Регистр MSTR_ARMISC1

Формат регистра MSTR_ARMISC1 приведен в следующей таблице:

Таблица 25.44. Формат регистра MSTR_ARMISC1

Разряды	Название	Описание		Сброс
31:24	-	Резерв	RO	0
23:0	mstr_armisc_info1	AXI master read misc information. Не является частью стандартного интерфейса AXI и служит как опционное дополнение к AXI интерфейсу. Соответствия битов: [15:0] – TLP requester ID [23:16] – TLP TAG	RO	0

25.2.1.41 Регистр MSTR_AMISC_LAST

Формат регистра MSTR_AMISC_LAST приведен в следующей таблице:

Таблица 25.45. Формат регистра MSTR_AMISC_LAST

Разряды	Название	Описание		Сброс
31:9	-	Резерв	RO	0
8	mstr_armisc_info_zeroread	AXI master read misc information. Не является частью стандартного интерфейса AXI и служит как опционное дополнение к AXI интерфейсу. Установка бита в единицу означает чтение нулевой длинны	RO	0
7:5	-	Резерв	RO	0
4	mstr_armisc_info_last	AXI master read request last TLP bit. Не является частью стандартного интерфейса AXI и служит как опционное дополнение к AXI интерфейсу. Установка бита в единицу означает последнюю TLP транзакцию в запросе чтения	RO	0
3:1	-	Резерв	RO	0
0	mstr_awmisc_info_last	AXI master write request last TLP bit. Не является частью стандартного интерфейса AXI и служит как опционное дополнение к AXI интерфейсу. Установка бита в единицу означает последнюю TLP транзакцию в запросе записи	RO	0

25.2.1.42 Регистр MSTR_AMISC_DMA

Формат регистра MSTR_AMISC_DMA приведен в следующей таблице:

Таблица 25.46. Формат регистра MSTR_AMISC_DMA

Разряды	Название	Описание		Сброс
31:14	-	Резерв	RO	0
13:8	mstr_armisc_info_dma	AXI master read request last TLP bit. Не является частью стандартного интерфейса AXI и служит как опционное дополнение к AXI интерфейсу. Установка бита в единицу означает последнюю TLP тразнакцию в запросе чтения	RO	0
7:6	-	Резерв	RO	0
5:0	mstr_awmisc_info_dma	AXI write master transaction’s DMA bits. Не является частью стандартного интерфейса AXI и служит как опционное дополнение к AXI интерфейсу. Биты [5:3] означают номер DMA канала Биты [2:0] означают тип DMA запроса 000 – Non-DMA request 001 – DMA write channel MRd request 010 – Non-DMA completion 011 – DMA write channel MWr request 100 – Reserved 101 – DMA write channel LL element MRd request 110 – Reserved 111 – DMA read channel LL element MRd request	RO	0

25.2.1.43 Регистр SLV_AWMISC

Формат регистра SLV_AWMISC приведен в следующей таблице:

Таблица 25.47. Формат регистра SLV_AWMISC

Разряды	Название	Описание		Сброс
31:24	slv_awmisc_info_p_tag	AXI slave write request Tag. Не является частью стандартного интерфейса AXI и служит как опционное дополнение к AXI интерфейсу.	RW	0
23:22	-	Резерв	RO	0
21:0	slv_awmisc_info	AXI slave write transaction associated misc information. Не является частью стандартного интерфейса AXI и служит как опционное дополнение к AXI интерфейсу. Соответствия битов: [4:0] – TLP TYPE [5] – Резерв [6] – TLP EP бит [7] – Резерв [8] – TLP NS бит [9] – TLP RO бит [12:10] – TLP TC биты [20:13] – TLP MSG код [21] – Резерв Резервные биты должны быть установлены в 0	RW	0

25.2.1.44 Регистр SLV_AWMISC_HDR0

Формат регистра SLV_AWMISC_HDR0 приведен в следующей таблице:

Таблица 25.48. Формат регистра SLV_AWMISC_HDR0

Разряды	Название	Описание		Сброс
31:0	slv_awmisc_info_hdr_3dw	AXI slave write misc information. Не является частью стандартного интерфейса AXI и служит как опционное дополнение к AXI интерфейсу. Содержит третье слово заголовка Msg/MsgD	RW	0

25.2.1.45 Регистр SLV_AWMISC_HDR1

Формат регистра SLV_AWMISC_HDR1 приведен в следующей таблице:

Таблица 25.49. Формат регистра SLV_AWMISC_HDR1

Разряды	Название	Описание		Сброс
31:0	slv_awmisc_info_hdr_4dw	AXI slave write misc information. Не является частью стандартного интерфейса AXI и служит как опционное дополнение к AXI интерфейсу. Содержит четвертое слово заголовка Msg/MsgD	RW	0

25.2.1.46 Регистр SLV_ARMISC

Формат регистра SLV_ARMISC приведен в следующей таблице:

Таблица 25.50. Формат регистра SLV_ARMISC

Разряды	Название	Описание		Сброс
31:22	-	Резерв	RO	0
21:0	slv_armisc_info	AXI slave read transaction associated misc information. Не является частью стандартного интерфейса AXI и служит как опционное дополнение к AXI интерфейсу. Соответствия битов: [4:0] – TLP TYPE [5] – Резерв [6] – TLP EP бит [7] – Резерв [8] – TLP NS бит [9] – TLP RO бит [12:10] – TLP TC биты [20:13] – TLP MSG код [21] – Резерв Резервные биты должны быть установлены в 0	RW	0

25.2.1.47 Регистр SLV_AMISC_ATU

Формат регистра SLV_AMISC_ATU приведен в следующей таблице:

Таблица 25.51. Формат регистра SLV_AMISC_ATU

Разряды	Название	Описание		Сброс
31:2	-	Резерв	RO	0
1	slv_armisc_info_atu_bypass	AXI slave read request internal ATU Bypass. Не является частью стандартного интерфейса AXI и служит как опционное дополнение к AXI интерфейсу. Если бит установлен, то запросы не будут обрабатываться во внутреннем ATU	RW	0
0	slv_awmisc_info_atu_bypass	AXI slave write request internal ATU Bypass. Не является частью стандартного интерфейса AXI и служит как опционное дополнение к AXI интерфейсу. Если бит установлен, то запросы не будут обрабатываться во внутреннем ATU	RW	0

25.2.1.48 Регистр SLV_MISC

Формат регистра SLV_MISC приведен в следующей таблице:

Таблица 25.52. Формат регистра SLV_MISC

Разряды	Название	Описание		Сброс
31:27	-	Резерв	RO	0
26:16	slv_rmisc_info	AXI slave read transaction responce associated misc information. Не является частью стандартного интерфейса AXI и служит как опционное дополнение к AXI интерфейсу. Соответствия битов:	RO	0

Разряды	Название	Описание		Сброс
		[0] – “Completion timeout” бит [1] – TLP EP бит [4:2] – TLP TC бит [5] – TLP NS бит [6] – TLP RO бит [9:7] – TLP completion status [10] – Non-posted request		
15:11	-	Резерв	RO	0
10:0	slv_bmisc_info	AXI slave write transaction responce associated misc information. Не является частью стандартного интерфейса AXI и служит как опционное дополнение к AXI интерфейсу. Соответствия битов: [0] – “Completion timeout” бит [1] – TLP EP бит [4:2] – TLP TC бит [5] – TLP NS бит [6] – TLP RO бит [9:7] – TLP completion status [10] – Non-posted request	RO	0

25.2.1.49 Регистр SLV_WMISC

Формат регистра SLV_WMISC приведен в следующей таблице:

Таблица 25.53. Формат регистра SLV_WMISC

Разряды	Название	Описание		Сброс
31:2	-	Резерв	RO	0
1	slv_wmisc_info_silentDrop	AXI Slave Write Data transaction related misc information. Не является частью стандартного интерфейса AXI и служит как опционное дополнение к AXI интерфейсу. Удаляет текущую транзакцию.	RW	0
0	slv_wmisc_info	AXI slave write data transaction related misc information. Не является частью стандартного интерфейса AXI и служит как опционное дополнение к AXI интерфейсу. Если установлен, то контроллер устанавливает “Poisoned TLP” (EP) бит в текущую и последующие транзакции записи данных	RW	0

25.2.1.50 Регистр sys_jesd_en

Формат регистра sys_jesd_en приведен в следующей таблице:

Таблица 25.54. Формат регистра sys_jesd_en

Разряды	Название	Описание		Сброс
31:1	-	Резерв	RO	0

Разряды	Название	Описание		Сброс
0	jesd_en	Выбор режима работы PCIe или JESD	RW	0

25.2.1.51 Регистр jesd_csr

Формат регистра jesd_csr приведен в следующей таблице:

Таблица 25.55. Формат регистра jesd_csr

Разряды	Название	Описание		Сброс
31	-	Резерв	RO	0
30:28	jesd_tx_clk_sel	Выбор источника формирования частоты передачи для JESD 000 - частота не выбрана 001 - mplla_word_clk 010 - dmplla_word_clk 010 - mplla_qword_clk 100 - mplla_oword_clk 101-111 - mplla_div_clk	RW	0
27:24	powerdown_busy	Сигнал подтверждения обработки запроса в PHY на изменение режима powerdown. 1 - запрос находится в обработке 0 - запрос выполнен Устанавливается в 1 при изменении значения поля powerdown. Сбрасывается аппаратно при выполнении перехода в запрашиваемый режим	RO	F
23:20	powerdown	Запрос на изменение режима powerdown	RW	2
19	-	Резерв	RO	0
18	en_tx_sample_reg	1 - tx_sample интерфейс берется с регистра 0 - tx_sample интерфейс идет комбинационно	RW	0
17	lmfc_div2_o		RO	0
16	lmfc_div2_i		RW	0
15:13	-	Резерв	RO	0
12	phy_ext_ctrl_sel	Разрешение использовать значения с регистров protocol0_ext_*	RW	0
11:10	jesd_rx_pstate	Установка power state приема- 2'b00: P0- Receiver is fully powered up and output receive clocks are active- 2'b01: P0S - RX voltage-controlled oscillator (VCO) is incontinuous calibration mode, output receive clocks are notavailable.- 2'b10: P1- RX analog front-end (AFE) and voltage regulators are powered up, but RX VCO is in reset- 2'b11: P2- RX LOS detector is powered up and the rest of RX is powered down	RW	0
9:8	jesd_tx_pstate	Установка power state передачи - 2'b00: P0-Transmitter is fully powered up. - 2'b01: P0S-Transmitter common mode is held, TX analog clocks are active, but TX serializer is off. - 2'b10: P1-Transmitter common mode is held, but TX analog	RW	0

Разряды	Название	Описание		Сброс
		clocks and TX serializer are off. - 2'b11: P2-Transmitter is powered down.		
7:6	jesd_rx_width	Ширина шины данных RX - 2'b00: 8-bit - 2'b01: 10-bit - 2'b10: 16-bit - 2'b11: 20-bit	RW	3
5:4	jesd_tx_width	Ширина шины данных TX - 2'b00: 8-bit - 2'b01: 10-bit - 2'b10: 16-bit - 2'b11: 20-bit	RW	3
3:2	jesd_rx_rate	Скорость приема данных RX - 2'b00: baud - 2'b01: baud/2 - 2'b10: baud/4 - 2'b11: baud/8	RW	0
1:0	jesd_tx_rate	Скорость передачи данных TX - 2'b00: baud - 2'b01: baud/2 - 2'b10: baud/4 - 2'b11: baud/8	RW	0

25.2.1.52 Регистр sram_crt

Формат регистра sram_crt приведен в следующей таблице:

Таблица 25.56. Формат регистра sram_crt

Разряды	Название	Описание		Сброс
31:10	-	Резерв	RO	0
9:8	phy_nominal_vph_sel	Выбор напряжения питания vph 2'b00 - Резерв 2'b01 - Резерв 2'b10 - 1.5В 2'b11 - 1.8В	RW	3
7:5	-	Резерв	RO	0
4	cr_para_sel	Выбор интерфейса доступа к регистрам PHY 1 - Control register 0 - JTAG	RW	1
3	-	Резерв	RO	0
2	sram_init_done	Инициализация sram завершена	RO	0
1	sram_ext_ld_done	Внешний доступ к sram завершен.Устанавливается програмно, после завершения работы с sram	RW	1

Разряды	Название	Описание		Сброс
0	sram_bypass	Выбор режима работы1 - bypass разрешен0 - bypass запрещенControl signal when asserted, bypasses the SRAM interface. In this case, the adaptation and calib	RW	1

25.2.1.53 Регистр protocol0_ext_bs

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_bs приведен в следующей таблице:

Таблица 25.57. Формат регистра protocol0_ext_bs

Разряды	Название	Описание		Сброс
31:9	-	Резерв	RO	0
8:4	protocol0_ext_bs_rx_level	TX/RX boundary scan swing/level settings Sets the boundary scan swing and level settings for the PHY	RW	0xB
3:2	-	Резерв	RO	0
1	protocol0_ext_bs_tx_lowswing	TX/RX boundary scan swing/level settings Sets the boundary scan swing and level settings for the PHY	RW	0
0	protocol0_ext_bs_rx_bigswing	TX/RX boundary scan swing/level settings Sets the boundary scan swing and level settings for the PHY	RW	0x1

25.2.1.54 Регистр protocol0_ext_bandwidth

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_bandwidth приведен в следующей таблице:

Таблица 25.58. Формат регистра protocol0_ext_bandwidth

Разряды	Название	Описание		Сброс
31:16	protocol0_ext_mpllb_bandwidth	MPLLB bandwidth control	RW	0xA036
15:0	protocol0_ext_mplla_bandwidth	MPLLA bandwidth control	RW	0xA036

25.2.1.55 Регистр protocol0_ext_mplla_div

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_mplla_div приведен в следующей таблице:

Таблица 25.59. Формат регистра protocol0_ext_mplla_div

Разряды	Название	Описание		Сброс
31:25	protocol0_ext_mplla_div_multiplier		RW	0
24:23	protocol0_ext_mplla_tx_clk_div		RW	0
22:12	protocol0_ext_mplla_fracn_ctrl		RW	0
11	protocol0_ext_mplla_div_clk_en		RW	0
10	protocol0_ext_mplla_div8_clk_en		RW	0x1
9	protocol0_ext_mplla_div16p5_clk_en		RW	0
8	protocol0_ext_mplla_div10_clk_en		RW	0
7:0	protocol0_ext_mplla_multiplier		RW	0x50

25.2.1.56 Регистр protocol0_ext_mplla_ssc

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_mplla_ssc приведен в следующей таблице:

Таблица 25.60. Формат регистра protocol0_ext_mplla_ssc

Разряды	Название	Описание		Сброс
31:22	-	резерс	RO	0
21	protocol0_ext_mplla_ssc_up_spread		RW	0
20	protocol0_ext_mplla_ssc_clk_sel		RW	0
19:12	protocol0_ext_mplla_ssc_freq_cnt_peak		RW	0x51
11:0	protocol0_ext_mplla_ssc_freq_cnt_init		RW	0x60

25.2.1.57 Регистр protocol0_ext_mpllb_div

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_mpllb_div приведен в следующей таблице:

Таблица 25.61. Формат регистра protocol0_ext_mpllb_div

Разряды	Название	Описание		Сброс
31:25	protocol0_ext_mpllb_div_multiplier		RW	0
24:23	protocol0_ext_mpllb_tx_clk_div		RW	0
22:12	protocol0_ext_mpllb_fracn_ctrl		RW	0
11	protocol0_ext_mpllb_div_clk_en		RW	0
10	protocol0_ext_mpllb_div8_clk_en		RW	0x1

Разряды	Название	Описание		Сброс
9	-	Резерв	RO	0
8	protocol0_ext_mpllb_div10_clk_en		RW	0
7:0	protocol0_ext_mpllb_multiplier		RW	0x50

25.2.1.58 Регистр protocol0_ext_mpllb_ssc

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_mpllb_ssc приведен в следующей таблице:

Таблица 25.62. Формат регистра protocol0_ext_mpllb_ssc

Разряды	Название	Описание		Сброс
31:22	-	Резерв	RO	0
21	protocol0_ext_mpllb_ssc_up_spread		RW	0
20	protocol0_ext_mpllb_ssc_clk_sel		RW	0
19:12	protocol0_ext_mpllb_ssc_freq_cnt_peak		RW	0x51
11:0	protocol0_ext_mpllb_ssc_freq_cnt_init		RW	0x60

25.2.1.59 Регистр protocol0_ext_ref_and_rx_adapt

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_ref_and_rx_adapt приведен в следующей таблице:

Таблица 25.63. Формат регистра protocol0_ext_ref_and_rx_adapt

Разряды	Название	Описание		Сброс
31:29	-	Резерв	RO	0
28:24	protocol0_ext_rx_vref_ctrl		RW	0x11
23	-	Резерв	RO	0
22	protocol0_ext_ref_clk_mpllb_div2_en		RW	0x1
21	protocol0_ext_ref_clk_mplla_div2_en		RW	0x1
20	protocol0_ext_ref_clk_div2_en		RW	0
19	-	Резерв	RO	0
18:16	protocol0_ext_ref_range		RW	0x3
15:12	protocol0_ext_rx_adapt_sel_g4		RW	0
11:8	protocol0_ext_rx_adapt_sel_g3		RW	0
7:4	protocol0_ext_rx_adapt_sel_g2		RW	0

Разряды	Название	Описание		Сброс
3:0	protocol0_ext_rx_adapt_sel_g1		RW	0

25.2.1.60 Регистр protocol0_ext_rx_adapt_mode

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_rx_adapt_mode приведен в следующей таблице:

Таблица 25.64. Формат регистра protocol0_ext_rx_adapt_mode

Разряды	Название	Описание		Сброс
31:24	protocol0_ext_rx_adapt_mode_g4		RW	0
23:16	protocol0_ext_rx_adapt_mode_g3		RW	0
15:8	protocol0_ext_rx_adapt_mode_g2		RW	0
7:0	protocol0_ext_rx_adapt_mode_g1		RW	0

25.2.1.61 Регистр protocol0_ext_rx_cdr_ppm_max_g1

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_rx_cdr_ppm_max_g1 приведен в следующей таблице:

Таблица 25.65. Формат регистра protocol0_ext_rx_cdr_ppm_max_g1

Разряды	Название	Описание		Сброс
31:20	-	Резерв	RO	0
19:0	protocol0_ext_rx_cdr_ppm_max_g1		RW	0x12

25.2.1.62 Регистр protocol0_ext_rx_cdr_ppm_max_g2

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_rx_cdr_ppm_max_g2 приведен в следующей таблице:

Таблица 25.66. Формат регистра protocol0_ext_rx_cdr_ppm_max_g2

Разряды	Название	Описание		Сброс
31:20	-	Резерв	RO	0
19:0	protocol0_ext_rx_cdr_ppm_max_g2		RW	0x12

25.2.1.63 Регистр protocol0_ext_rx_cdr_ppm_max_g3

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_rx_cdr_ppm_max_g3 приведен в следующей таблице:

Таблица 25.67. Формат регистра protocol0_ext_rx_cdr_ppm_max_g3

Разряды	Название	Описание		Сброс
31:20	-	Резерв	RO	0
19:0	protocol0_ext_rx_cdr_ppm_max_g3		RW	0x12

25.2.1.64 Регистр protocol0_ext_rx_cdr_ppm_max_g4

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_rx_cdr_ppm_max_g4 приведен в следующей таблице:

Таблица 25.68. Формат регистра protocol0_ext_rx_cdr_ppm_max_g4

Разряды	Название	Описание		Сброс
31:20	-	Резерв	RO	0
19:0	protocol0_ext_rx_cdr_ppm_max_g4		RW	0x12

25.2.1.65 Регистр protocol0_ext_rx_cdr_vco_freqband

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_rx_cdr_vco_freqband приведен в следующей таблице:

Таблица 25.69. Формат регистра protocol0_ext_rx_cdr_vco_freqband

Разряды	Название	Описание		Сброс
31:24	protocol0_ext_rx_cdr_vco_freqband_g4		RW	0
23:16	protocol0_ext_rx_cdr_vco_freqband_g3		RW	0
15:8	protocol0_ext_rx_cdr_vco_freqband_g2		RW	0
7:0	protocol0_ext_rx_cdr_vco_freqband_g1		RW	0

25.2.1.66 Регистр protocol0_ext_rx_cdr_vco

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_rx_cdr_vco приведен в следующей таблице:

Таблица 25.70. Формат регистра protocol0_ext_rx_cdr_vco

Разряды	Название	Описание		Сброс
31:28	protocol0_ext_rx_cdr_vco_temp_comp_en_g4		RW	0
27:24	protocol0_ext_rx_cdr_vco_temp_comp_en_g3		RW	0
23:20	protocol0_ext_rx_cdr_vco_temp_comp_en_g2		RW	0
19:16	protocol0_ext_rx_cdr_vco_temp_comp_en_g1		RW	0

Разряды	Название	Описание		Сброс
15:12	protocol0_ext_rx_cdr_vco_step_ctrl_g4		RW	0
11:8	protocol0_ext_rx_cdr_vco_step_ctrl_g3		RW	0
7:4	protocol0_ext_rx_cdr_vco_step_ctrl_g2		RW	0
3:0	protocol0_ext_rx_cdr_vco_step_ctrl_g1		RW	0

25.2.1.67 Регистр protocol0_ext_rx_delta_iq_g12

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_rx_delta_iq_g12 приведен в следующей таблице:

Таблица 25.71. Формат регистра protocol0_ext_rx_delta_iq_g12

Разряды	Название	Описание		Сброс
31:16	protocol0_ext_rx_delta_iq_g2		RW	0
15:0	protocol0_ext_rx_delta_iq_g1		RW	0

25.2.1.68 Регистр protocol0_ext_rx_delta_iq_g34

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_rx_delta_iq_g34 приведен в следующей таблице:

Таблица 25.72. Формат регистра protocol0_ext_rx_delta_iq_g34

Разряды	Название	Описание		Сброс
31:16	protocol0_ext_rx_delta_iq_g4		RW	0
15:0	protocol0_ext_rx_delta_iq_g3		RW	0

25.2.1.69 Регистр protocol0_ext_rx_dfe_bypass_and_att_lvl_g12

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_rx_dfe_bypass_and_att_lvl_g12 приведен в следующей таблице:

Таблица 25.73. Формат регистра protocol0_ext_rx_dfe_bypass_and_att_lvl_g12

Разряды	Название	Описание		Сброс
31:20	protocol0_ext_rx_eq_att_lvl_g2		RW	0
19:8	protocol0_ext_rx_eq_att_lvl_g1		RW	0
7:4	protocol0_ext_rx_dfe_bypass_g2		RW	0
3:0	protocol0_ext_rx_dfe_bypass_g1		RW	0

25.2.1.70 Регистр protocol0_ext_rx_dfe_bypass_and_att_lvl_g34

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_rx_dfe_bypass_and_att_lvl_g34 приведен в следующей таблице:

Таблица 25.74. Формат регистра protocol0_ext_rx_dfe_bypass_and_att_lvl_g34

Разряды	Название	Описание		Сброс
31:20	protocol0_ext_rx_eq_att_lvl_g4		RW	0
19:8	protocol0_ext_rx_eq_att_lvl_g3		RW	0
7:4	protocol0_ext_rx_dfe_bypass_g4		RW	0
3:0	protocol0_ext_rx_dfe_bypass_g3		RW	0

25.2.1.71 Регистр protocol0_ext_rx_eq_ctle_boost_g1

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_rx_eq_ctle_boost_g1 приведен в следующей таблице:

Таблица 25.75. Формат регистра protocol0_ext_rx_eq_ctle_boost_g1

Разряды	Название	Описание		Сброс
31:20	-	Резерв	RO	0
19:0	protocol0_ext_rx_eq_ctle_boost_g1		RW	0

25.2.1.72 Регистр protocol0_ext_rx_eq_ctle_boost_g2

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_rx_eq_ctle_boost_g2 приведен в следующей таблице:

Таблица 25.76. Формат регистра protocol0_ext_rx_eq_ctle_boost_g2

Разряды	Название	Описание		Сброс
31:20	-	Резерв	RO	0
19:0	protocol0_ext_rx_eq_ctle_boost_g2		RW	0

25.2.1.73 Регистр protocol0_ext_rx_eq_ctle_boost_g3

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_rx_eq_ctle_boost_g3 приведен в следующей таблице:

Таблица 25.77. Формат регистра protocol0_ext_rx_eq_ctle_boost_g3

Разряды	Название	Описание		Сброс
31:20	-	Резерв	RO	0
19:0	protocol0_ext_rx_eq_ctle_boost_g3		RW	0

25.2.1.74 Регистр protocol0_ext_rx_eq_ctle_boost_g4

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_rx_eq_ctle_boost_g4 приведен в следующей таблице:

Таблица 25.78. Формат регистра protocol0_ext_rx_eq_ctle_boost_g4

Разряды	Название	Описание		Сброс
31:20	-	Резерв	RO	0
19:0	protocol0_ext_rx_eq_ctle_boost_g4		RW	0

25.2.1.75 Регистр protocol0_ext_rx_eq_ctle_pole

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_rx_eq_ctle_pole приведен в следующей таблице:

Таблица 25.79. Формат регистра protocol0_ext_rx_eq_ctle_pole

Разряды	Название	Описание		Сброс
31:24	protocol0_ext_rx_eq_ctle_pole_g4		RW	0
23:16	protocol0_ext_rx_eq_ctle_pole_g3		RW	0
15:8	protocol0_ext_rx_eq_ctle_pole_g2		RW	0
7:0	protocol0_ext_rx_eq_ctle_pole_g1		RW	0

25.2.1.76 Регистр protocol0_ext_rx_eq_dfe_tap1_g1

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_rx_eq_dfe_tap1_g1 приведен в следующей таблице:

Таблица 25.80. Формат регистра protocol0_ext_rx_eq_dfe_tap1_g1

Разряды	Название	Описание		Сброс
31:0	protocol0_ext_rx_eq_dfe_tap1_g1		RW	0

25.2.1.77 Регистр protocol0_ext_rx_eq_dfe_tap1_g2

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_rx_eq_dfe_tap1_g2 приведен в следующей таблице:

Таблица 25.81. Формат регистра protocol0_ext_rx_eq_dfe_tap1_g2

Разряды	Название	Описание		Сброс
31:0	protocol0_ext_rx_eq_dfe_tap1_g2		RW	0

25.2.1.78 Регистр protocol0_ext_rx_eq_dfe_tap1_g3

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_rx_eq_dfe_tap1_g3 приведен в следующей таблице:

Таблица 25.82. Формат регистра protocol0_ext_rx_eq_dfe_tap1_g3

Разряды	Название	Описание		Сброс
31:0	protocol0_ext_rx_eq_dfe_tap1_g3		RW	0

25.2.1.79 Регистр protocol0_ext_rx_eq_dfe_tap1_g4

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_rx_eq_dfe_tap1_g4 приведен в следующей таблице:

Таблица 25.83. Формат регистра protocol0_ext_rx_eq_dfe_tap1_g4

Разряды	Название	Описание		Сброс
31:0	protocol0_ext_rx_eq_dfe_tap1_g4		RW	0

25.2.1.80 Регистр protocol0_ext_rx_eq_vga1_gain_g12

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_rx_eq_vga1_gain_g12 приведен в следующей таблице:

Таблица 25.84. Формат регистра protocol0_ext_rx_eq_vga1_gain_g12

Разряды	Название	Описание		Сброс
31:28	-	Резерв	RO	0
27:16	protocol0_ext_rx_eq_vga1_gain_g2		RW	0
15:12	-	Резерв	RO	0
11:0	protocol0_ext_rx_eq_vga1_gain_g1		RW	0

25.2.1.81 Регистр protocol0_ext_rx_eq_vga1_gain_g34

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_rx_eq_vga1_gain_g34 приведен в следующей таблице:

Таблица 25.85. Формат регистра protocol0_ext_rx_eq_vga1_gain_g34

Разряды	Название	Описание		Сброс
31:28	-	Резерв	RO	0
27:16	protocol0_ext_rx_eq_vga1_gain_g4		RW	0
15:12	-	Резерв	RO	0
11:0	protocol0_ext_rx_eq_vga1_gain_g3		RW	0

25.2.1.82 Регистр protocol0_ext_rx_eq_vga2_gain_g12

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_rx_eq_vga2_gain_g12 приведен в следующей таблице:

Таблица 25.86. Формат регистра protocol0_ext_rx_eq_vga2_gain_g12

Разряды	Название	Описание		Сброс
31:28	-	Резерв	RO	0
27:16	protocol0_ext_rx_eq_vga2_gain_g2		RW	0
15:12	-	Резерв	RO	0
11:0	protocol0_ext_rx_eq_vga2_gain_g1		RW	0

25.2.1.83 Регистр protocol0_ext_rx_eq_vga2_gain_g34

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_rx_eq_vga2_gain_g34 приведен в следующей таблице:

Таблица 25.87. Формат регистра protocol0_ext_rx_eq_vga2_gain_g34

Разряды	Название	Описание		Сброс
31:28	-	Резерв	RO	0
27:16	protocol0_ext_rx_eq_vga1_gain_g4		RW	0
15:12	-	Резерв	RO	0
11:0	protocol0_ext_rx_eq_vga2_gain_g3		RW	0

25.2.1.84 Регистр protocol0_ext_rx_los

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_rx_los приведен в следующей таблице:

Таблица 25.88. Формат регистра protocol0_ext_rx_los

Разряды	Название	Описание		Сброс
31:25	-	Резерв	RO	0
24	protocol0_ext_rx_los_lfps_en		RW	0
23	-	Резерв	RO	0
22:12	protocol0_ext_rx_los_pwr_up_cnt		RW	0
11:0	protocol0_ext_rx_los_threshold		RW	0

25.2.1.85 Регистр protocol0_ext_rx_misc_g1

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_rx_misc_g1 приведен в следующей таблице:

Таблица 25.89. Формат регистра protocol0_ext_rx_misc_g1

Разряды	Название	Описание		Сброс
31:0	protocol0_ext_rx_misc_g1		RW	0

25.2.1.86 Регистр protocol0_ext_rx_misc_g2

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_rx_misc_g2 приведен в следующей таблице:

Таблица 25.90. Формат регистра protocol0_ext_rx_misc_g2

Разряды	Название	Описание		Сброс
31:0	protocol0_ext_rx_misc_g2		RW	0

25.2.1.87 Регистр protocol0_ext_rx_misc_g3

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_rx_misc_g3 приведен в следующей таблице:

Таблица 25.91. Формат регистра protocol0_ext_rx_misc_g3

Разряды	Название	Описание		Сброс
31:0	protocol0_ext_rx_misc_g3		RW	0

25.2.1.88 Регистр protocol0_ext_rx_misc_g4

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_rx_misc_g4 приведен в следующей таблице:

Таблица 25.92. Формат регистра protocol0_ext_rx_misc_g4

Разряды	Название	Описание		Сброс
31:0	protocol0_ext_rx_misc_g4		RW	0

25.2.1.89 Регистр protocol0_ext_rx_ref_ld_val

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_rx_ref_ld_val приведен в следующей таблице:

Таблица 25.93. Формат регистра protocol0_ext_rx_ref_ld_val

Разряды	Название	Описание		Сброс
31	-	Резерв	RO	0
30:24	protocol0_ext_rx_ref_ld_val_g4		RW	0x22
23	-	Резерв	RO	0
22:16	protocol0_ext_rx_ref_ld_val_g3		RW	0x22
15	-	Резерв	RO	0
14:8	protocol0_ext_rx_ref_ld_val_g2		RW	0x22
7	-	Резерв	RO	0
6:0	protocol0_ext_rx_ref_ld_val_g1		RW	0x22

25.2.1.90 Регистр protocol0_ext_rx_vco_ld_val_g12

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_rx_vco_ld_val_g12 приведен в следующей таблице:

Таблица 25.94. Формат регистра protocol0_ext_rx_vco_ld_val_g12

Разряды	Название	Описание		Сброс
31:29	-	Резерв	RO	0
28:16	protocol0_ext_rx_vco_ld_val_g2		RW	0x550
15:13	-	Резерв	RO	0
12:0	protocol0_ext_rx_vco_ld_val_g1		RW	0x550

25.2.1.91 Регистр protocol0_ext_rx_vco_ld_val_g34

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_rx_vco_ld_val_g34 приведен в следующей таблице:

Таблица 25.95. Формат регистра protocol0_ext_rx_vco_ld_val_g34

Разряды	Название	Описание		Сброс
31:29	-	Резерв	RO	0
28:16	protocol0_ext_rx_vco_ld_val_g4		RW	0x550
15:13	-	Резерв	RO	0
12:0	protocol0_ext_rx_vco_ld_val_g3		RW	0x550

25.2.1.92 Регистр protocol0_ext_sup_misc

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_sup_misc приведен в следующей таблице:

Таблица 25.96. Формат регистра protocol0_ext_sup_misc

Разряды	Название	Описание		Сброс
31:24	protocol0_ext_sup_misc_g4		RW	0x3
23:16	protocol0_ext_sup_misc_g3		RW	0x3
15:8	protocol0_ext_sup_misc_g2		RW	0x3
7:0	protocol0_ext_sup_misc_g1		RW	0x3

25.2.1.93 Регистр protocol0_ext_tx_eq_main_g1

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_tx_eq_main_g1 приведен в следующей таблице:

Таблица 25.97. Формат регистра protocol0_ext_tx_eq_main_g1

Разряды	Название	Описание		Сброс
31:24	-	Резерв	RO	0
23:0	protocol0_ext_tx_eq_main_g1		RW	0

25.2.1.94 Регистр protocol0_ext_tx_eq_main_g2

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_tx_eq_main_g2 приведен в следующей таблице:

Таблица 25.98. Формат регистра protocol0_ext_tx_eq_main_g2

Разряды	Название	Описание		Сброс
31:24	-	Резерв	RO	0
23:0	protocol0_ext_tx_eq_main_g2		RW	0

25.2.1.95 Регистр protocol0_ext_tx_eq_main_g3

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_tx_eq_main_g3 приведен в следующей таблице:

Таблица 25.99. Формат регистра protocol0_ext_tx_eq_main_g3

Разряды	Название	Описание		Сброс
31:24	-	Резерв	RO	0
23:0	protocol0_ext_tx_eq_main_g3		RW	0

25.2.1.96 Регистр protocol0_ext_tx_eq_main_g4

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_tx_eq_main_g4 приведен в следующей таблице:

Таблица 25.100. Формат регистра protocol0_ext_tx_eq_main_g4

Разряды	Название	Описание		Сброс
31:24	-	Резерв	RO	0
23:0	protocol0_ext_tx_eq_main_g4		RW	0

25.2.1.97 Регистр protocol0_ext_tx_eq_ovrd

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_tx_eq_ovrd приведен в следующей таблице:

Таблица 25.101. Формат регистра protocol0_ext_tx_eq_ovrd

Разряды	Название	Описание		Сброс
31:16	protocol0_ext_tx_iboost_lvl		RW	0
15:12	protocol0_ext_tx_eq_ovrd_g4		RW	0
11:8	protocol0_ext_tx_eq_ovrd_g3		RW	0
7:4	protocol0_ext_tx_eq_ovrd_g2		RW	0
3:0	protocol0_ext_tx_eq_ovrd_g1		RW	0

25.2.1.98 Регистр protocol0_ext_tx_eq_post_g1

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_tx_eq_post_g1 приведен в следующей таблице:

Таблица 25.102. Формат регистра protocol0_ext_tx_eq_post_g1

Разряды	Название	Описание		Сброс
31:20	-	Резерв	RO	0
19:0	protocol0_ext_tx_eq_post_g1		RW	0

25.2.1.99 Регистр protocol0_ext_tx_eq_post_g2

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_tx_eq_post_g2 приведен в следующей таблице:

Таблица 25.103. Формат регистра protocol0_ext_tx_eq_post_g2

Разряды	Название	Описание		Сброс
31:20	-	Резерв	RO	0
19:0	protocol0_ext_tx_eq_post_g2		RW	0

25.2.1.100 Регистр protocol0_ext_tx_eq_post_g3

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_tx_eq_post_g3 приведен в следующей таблице:

Таблица 25.104. Формат регистра protocol0_ext_tx_eq_post_g3

Разряды	Название	Описание		Сброс
31:20	-	Резерв	RO	0
19:0	protocol0_ext_tx_eq_post_g3		RW	0

25.2.1.101 Регистр protocol0_ext_tx_eq_post_g4

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_tx_eq_post_g4 приведен в следующей таблице:

Таблица 25.105. Формат регистра protocol0_ext_tx_eq_post_g4

Разряды	Название	Описание		Сброс
31:20	-	Резерв	RO	0
19:0	protocol0_ext_tx_eq_post_g4		RW	0

25.2.1.102 Регистр protocol0_ext_tx_eq_pre_g1

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_tx_eq_pre_g1 приведен в следующей таблице:

Таблица 25.106. Формат регистра protocol0_ext_tx_eq_pre_g1

Разряды	Название	Описание		Сброс
31:20	-	Резерв	RO	0
19:0	protocol0_ext_tx_eq_pre_g1		RW	0

25.2.1.103 Регистр protocol0_ext_tx_eq_pre_g2

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_tx_eq_pre_g2 приведен в следующей таблице:

Таблица 25.107. Формат регистра protocol0_ext_tx_eq_pre_g2

Разряды	Название	Описание		Сброс
31:20	-	Резерв	RO	0
19:0	protocol0_ext_tx_eq_pre_g2		RW	0

25.2.1.104 Регистр protocol0_ext_tx_eq_pre_g3

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_tx_eq_pre_g3 приведен в следующей таблице:

Таблица 25.108. Формат регистра protocol0_ext_tx_eq_pre_g3

Разряды	Название	Описание		Сброс
31:20	-	Резерв	RO	0
19:0	protocol0_ext_tx_eq_pre_g3		RW	0

25.2.1.105 Регистр protocol0_ext_tx_eq_pre_g4

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_tx_eq_pre_g4 приведен в следующей таблице:

Таблица 25.109. Формат регистра protocol0_ext_tx_eq_pre_g4

Разряды	Название	Описание		Сброс
31:20	-	Резерв	RO	0
19:0	protocol0_ext_tx_eq_pre_g4		RW	0

25.2.1.106 Регистр protocol0_ext_tx_misc_g1

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_tx_misc_g1 приведен в следующей таблице:

Таблица 25.110. Формат регистра protocol0_ext_tx_misc_g1

Разряды	Название	Описание		Сброс
31:0	protocol0_ext_tx_misc_g1		RW	0

25.2.1.107 Регистр protocol0_ext_tx_misc_g2

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_tx_misc_g2 приведен в следующей таблице:

Таблица 25.111. Формат регистра protocol0_ext_tx_misc_g2

Разряды	Название	Описание		Сброс
31:0	protocol0_ext_tx_misc_g2		RW	0

25.2.1.108 Регистр protocol0_ext_tx_misc_g3

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_tx_misc_g3 приведен в следующей таблице:

Таблица 25.112. Формат регистра protocol0_ext_tx_misc_g3

Разряды	Название	Описание		Сброс
31:0	protocol0_ext_tx_misc_g3		RW	0

25.2.1.109 Регистр protocol0_ext_tx_misc_g4

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_tx_misc_g4 приведен в следующей таблице:

Таблица 25.113. Формат регистра protocol0_ext_tx_misc_g4

Разряды	Название	Описание		Сброс
31:0	protocol0_ext_tx_misc_g4		RW	0

25.2.1.110 Регистр protocol0_ext_tx_vboost_lvl

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_tx_vboost_lvl приведен в следующей таблице:

Таблица 25.114. Формат регистра protocol0_ext_tx_vboost_lvl

Разряды	Название	Описание		Сброс
31:3	-	Резерв	RO	0
2:0	protocol0_ext_tx_vboost_lvl		RW	0x5

25.2.1.111 Регистр protocol0_ext_sup_misc_recal_bank

Поля данного регистра соответствуют одноименным входам контроллера PHY

Формат регистра protocol0_ext_sup_misc_recal_bank приведен в следующей таблице:

Таблица 25.115. Формат регистра protocol0_ext_sup_misc_recal_bank

Разряды	Название	Описание		Сброс
31:9	-	Резерв	RO	0
8	protocol_ext_sup_misc_ovrd_en		RW	0
7	-	Резерв	RO	0
6	protocol_ext_mpllb_recal_bank_sel_ovrd_en		RW	0
5:4	protocol_ext_mpllb_recal_bank_sel		RW	0
3	-	Резерв	RO	0
2	protocol_ext_mplla_recal_bank_sel_ovrd_en		RW	0
1:0	protocol_ext_mplla_recal_bank_sel		RW	0

25.2.1.112 Регистр rst_status

Формат регистра rst_status приведен в следующей таблице:

Таблица 25.116. Формат регистра rst_status

Разряды	Название	Описание		Сброс
31:10	-	Резерв	RO	0
9	mstr_axi_rst_n	После снятия mstr_axi_rst_n сброса аппаратно устанавливается в 1	RO	0
8	slv_axi_rst_n	После снятия slv_axi_rst_n сброса аппаратно устанавливается в 1	RO	0
7	dbi_axi_rst_n	После снятия dbi_axi_rst_n сброса аппаратно устанавливается в 1	RO	0
6	phy_reg_rst_n	После снятия phy_reg_rst_n сброса аппаратно устанавливается в 1	RO	0
5	phy_rst_n	После снятия phy_rst_n сброса аппаратно устанавливается в 1	RO	0
4	non-sticky_rst_n	После снятия non-sticky_rst_n сброса аппаратно устанавливается в 1	RO	0
3	sticky_rst_n	После снятия sticky_rst_n сброса аппаратно устанавливается в 1	RO	0
2	core_rst_n	После снятия core_rst_n сброса аппаратно устанавливается в 1	RO	0
1	pwr_rst_n	После снятия pwr_rst_n сброса аппаратно устанавливается в 1	RO	0
0	sync_perst_n	После снятия sync_perst_n сброса аппаратно устанавливается в 1	RO	0

25.2.2 Регистры блока JESD_TX

Перечень регистров приведен в Таблица 25.117.

Таблица 25.117. Перечень программно-доступных регистров блока JESD_TX

Условное Обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
BID	Bank ID	RW	0x0	0xC00
DID	Device ID	RW	0x0	0xC04
LID	Lane ID for lanes 0 to 3	RW	0x18820	0xC08
SUBCLASS	SUBCLASS Control	RW	0x1	0xC1C
CF	CF Control	RW	0x0	0x020
CS	CS Control	RW	0x0	0x024
F	F Control	RW	0x0	0x028
K	K Control	RW	0x10	0x02C
L	L Control	RW	0x3	0x030
N	N Control	RW	0xF	0x034
M	M Control	RW	0x3	0x038
NTOTAL	NTOTAL Control	RW	0xF	0x03C
S	S Control	RW	0x0	0x040
HD	HD Control	RW	0x0	0x044
SCR	SCR Control	RW	0x0	0x048
ENABLEMODULE	ENABLEMODULE Control	RW	0x0	0x050
ENABLELINKTEST	ENABLELINKTEST Control	RW	0x0	0x054
TAILBITS	TAILBITS Control	RW	0x0	0x058
LINKTEST_CUSTOM_DATA	LINKTEST_CUSTOM_DATA Control	RW	0x0	0x05C
LINKTEST_CUSTOM_ENABLE	LINKTEST_CUSTOM_ENABLE Control	RW	0x0	0x060
TX_FEATURE_DISABLE	TX_FEATURE_DISABLE Control	RW	0x0	0x120
TX_CORE_INTERRUPT_MASK	TX_CORE_INTERRUPT_MASK Control	RW	0xF	0x124
TX_CORE_INTERRUPT_STATUS	TX_CORE_INTERRUPT_STATUS Control	RC	0x0	0x128
TX_ILAS_AND_MISC_DETAILS	TX_ILAS_AND_MISC_DETAILS Control	RW	0x3000011	0x12C
PRBS_TEST_ENABLE	PRBS_TEST_ENABLE Control	RW	0x0	0x134
JTSPAT_TEST_ENABLE	JTSPAT_TEST_ENABLE Control	RW	0x0	0x138
CLK_RATIO	CLK_RATIO Control	RW	0x0	0x13C
TX_ILAS_DELAY	TX_ILAS_DELAY Control	RW	0x4	0x2C0
SOFT_SYNC	SOFT_SYNC Control	RW	0x0	0x2D0
TL_TEST_MODE	TL_TEST_MODE Control	RW	0x0	0x2DC
LMFC_DELAY	LMFC_DELAY Control	RW	0x0	0x2E0
SYSREF_MAXSHIFT	SYSREF_MAXSHIFT Control	RW	0x0	0x2E4

25.2.2.1 Регистр BID

Формат регистра BID приведен в Таблица 25.118.

Таблица 25.118. Формат регистра BID

Номер бита	Условное обозначение	Назначение	Сброс
31:4	–	Резерв	?
3:0	BID	Bank ID	0

25.2.2.2 Регистр DID

Формат регистра DID приведен в Таблица 25.119

Таблица 25.119. Формат регистра DID

Номер бита	Условное обозначение	Назначение	Сброс
31:4	–	Резерв	?
3:0	DID	Device ID	0

25.2.2.3 Регистр LID

Lane ID для полос от 0 до 3.

Определяет соотношение физической и логической полос. Например, если ‘h01 расположен в первых пяти битах [4: 0], соответствующих LID0, это означает, что канальный уровень пометит в метаданных логическую линию 0 как физическую линию 1. В отличие от Rx, это не изменит положение данных.

Формат регистра LID приведен в Таблица 25.120

Таблица 25.120. Формат регистра LID

Номер бита	Условное обозначение	Назначение	Сброс
31:20	–	Резерв	?
19:15	LID3	Определяет соотношение физической и логической полос.	3
14:10	LID2	Определяет соотношение физической и логической полос.	2
9:5	LID1	Определяет соотношение физической и логической полос.	1
4:0	LID0	Определяет соотношение физической и логической полос.	0

25.2.2.4 Регистр SUBCLASS

Выбор режима работы – Subclass 0, 1, или 2.

Формат регистра SUBCLASS приведен в Таблица 25.121.

Таблица 25.121. Формат регистра SUBCLASS

Номер бита	Условное Обозначение	Назначение	Сброс
31:3	–	Резерв	?
2:0	SUBCLASS	Выбор режима работы – Subclass 0, 1, или 2.	1

25.2.2.5 Регистр CF

Формат регистра CF приведен в Таблица 25.122.

Таблица 25.122. Формат регистра CF

Номер бита	Условное обозначение	Назначение	Сброс
31:18	–	Резерв	?
17:10	CF_FIRST_NIBBLE	Если CF = 1, это указывает на первую 4-битную позицию полубайта контрольных бит CF-lane	0
9:5	CF_FIRST_LANE	Указывает полосу, когда контрольные слова найдены	0
4:2	–	Резерв	?
1:0	CF	Количество контрольных слов на конвертер. (CF=0, or 1)	0

25.2.2.6 Регистр CS

Количество контрольных бит на семпл.

Формат регистра CS приведен в Таблица 25.25.123

Таблица 25.25.123. Формат регистра CS

Номер бита	Условное обозначение	Назначение	Сброс
31:2	–	Резерв	?
1:0	CS	Количество контрольных бит на семпл	0

25.2.2.7 Регистр F

Количество октетов в кадре. Допустимые значения: 0-7. Кодировка - двоичное значение минус 1.

Формат регистра F приведен в Таблица 25.124

Таблица 25.124. Формат регистра F

Номер бита	Условное обозначение	Назначение	Сброс
31:8	–	Резерв	?
7:0	F	Количество октетов в кадре	0

25.2.2.8 Регистр K

Количество кадров в мультикадре. Допустимые значения: 0-31. Кодировка - двоичное значение минус 1

Формат регистра K приведен в Таблица 25.125

Таблица 25.125. Формат регистра K

Номер бита	Условное обозначение	Назначение	Сброс
31:5	–	Резерв	?
4:0	K	Количество кадров в мультикадре.	5'h10

25.2.2.9 Регистр L

Количество полос на конвертер. Допустимые значения: 0-3. Кодировка - двоичное значение минус 1.

Формат регистра L приведен в Таблица 25.126

Таблица 25.126. Формат регистра L

Номер бита	Условное обозначение	Назначение	Сброс
31:5	–	Резерв	?
4:0	L	Количество полос на конвертер	5'h3

25.2.2.10 Регистр N

Разрешение конвертера. Допустимые значения: 7, 9, 11, 13, 14 и 15. Кодировка - двоичные значения минус 1.

Формат регистра N приведен в Таблица 25.127.

Таблица 25.127. Формат регистра N

Номер бита	Условное обозначение	Назначение	Сброс
31:5	–	Резерв	?
4:0	N	Разрешение конвертера	5'hF

25.2.2.11 Регистр M

Количество конвертеров на устройство. Допустимые значения: 0-7. Кодировка - двоичное значение минус 1.

Формат регистра M приведен в Таблица 25.128.

Таблица 25.128. Формат регистра M

Номер бита	Условное обозначение	Назначение	Сброс
31:8	–	Резерв	?
7:0	M	Количество конвертеров на устройство	8'h3

25.2.2.12 Регистр NTOTAL

Общее количество бит на семпл (N'). Должно быть целым числом, кратным 4. Допустимые значения: 7, 11, 15. Кодировка - двоичное значение минус 1.

Формат регистра NTOTAL приведен в Таблица 25.129.

Таблица 25.129. Формат регистра NTOTAL

Номер бита	Условное обозначение	Назначение	Сброс
31:5	–	Резерв	?

Номер бита	Условное обозначение	Назначение	Сброс
4:0	NTOTAL	Общее количество бит на семпл (N')	5'hF

25.2.2.13 Регистр S

Количество семплов на конвертер на цикл кадра. Допустимые значения: от 0 до 3. Кодировка - двоичное значение минус 1.

Формат регистра S приведен в Таблица 25.130.

Таблица 25.130. Формат регистра S

Номер бита	Условное обозначение	Назначение	Сброс
31:5	–	Резерв	?
4:0	S	Количество семплов на конвертер на цикл кадра	0

25.2.2.14 Регистр HD

Формат высокой плотности (HD).

Формат регистра HD приведен в Таблица 25.131.

Таблица 25.131. Формат регистра HD

Номер бита	Условное обозначение	Назначение	Сброс
31:1	–	Резерв	?
0	HD	Формат высокой плотности (HD) 0 – HD не используется 1 – HD используется.	0

25.2.2.15 Регистр SCR

Регистр управления скремблированием.

Формат регистра SCR приведен в Таблица 25.132.

Таблица 25.132. Формат регистра SCR

Номер бита	Условное обозначение	Назначение	Сброс
31:1	–	Резерв	?
0	SCR	Включение скремблирования	0

25.2.2.16 Регистр ENABLEMODULE

Регистр включения работы модуля

Формат регистра ENABLEMODULE приведен в Таблица 25.133.

Таблица 25.133. Формат регистра ENABLEMODULE

Номер бита	Условное обозначение	Назначение	Сброс
31:1	–	Резерв	?
0	ENABLEMODULE	0 – модуль отключен 1 – модуль включен.	0

25.2.2.17 Регистр ENABLELINKTEST

Регистр включения тестирование канального уровня.

Формат регистра ENABLELINKTEST приведен в Таблица 25.134.

Таблица 25.134. Формат регистра ENABLELINKTEST

Номер бита	Условное обозначение	Назначение	Сброс
31:2	–	Резерв	?
1:0	ENABLELINKTEST	0 – тест на канальном уровне не выполнен. 1 – передача непрерывной последовательности символов /K28.5/ (LINKTEST1). 2 – передача последовательности кодовой группы с последующей повторной последовательностью выравнивания дорожек (LINKTEST2). 3 – тест на канальном уровне не выполнен	0

25.2.2.18 Регистр TAILBITS

Регистр управления вставкой хвостовых бит.

Формат регистра TAILBITS приведен в Таблица 25.135.

Таблица 25.135. Формат регистра TAILBITS

Номер бита	Условное обозначение	Назначение	Сброс
31:2	–	Резерв	?
1:0	TAILBITS	0 – вставить ‘0’ 1 – вставить ‘1’ 2 – вставить хвостовой бит, сгенерированный из LFSR (или другой конструкции), который обеспечивает низкое содержание DC 3 – неверно (вставить «0»).	0

25.2.2.19 Регистр LINKTEST_CUSTOM_DATA

Регистр определяющий 16-битное слово, которое может передаваться непрерывно для тестирования канального уровня.

Формат регистра LINKTEST_CUSTOM_DATA приведен в Таблица 25.136.

Таблица 25.136. Формат регистра LINKTEST_CUSTOM_DATA

Номер бита	Условное обозначение	Назначение	Сброс
31:16	–	Резерв	?
15:0	LINKTEST_CUSTOM_DATA	16-битное слово, которое может передаваться непрерывно для тестирования канального уровня.	0

25.2.2.20 Регистр LINKTEST_CUSTOM_ENABLE

Регистр включения продолжительной передачи значения из LINKTEST_CUSTOM_DATA.

Формат регистра LINKTEST_CUSTOM_ENABLE приведен в Таблица 25.137.

Таблица 25.137. Формат регистра LINKTEST_CUSTOM_ENABLE

Номер бита	Условное обозначение	Назначение	Сброс
31:1	–	Резерв	?
0	LINKTEST_CUSTOM_ENABLE	Включить продолжительной передачи значения из LINKTEST_CUSTOM_DATA	0

25.2.2.21 Регистр TX_FEATURE_DISABLE

Регистр отключения функций модуля Tx для совместимости с другими устройствами

Формат регистра TX_FEATURE_DISABLE приведен в Таблица 25.138.

Таблица 25.138. Формат регистра TX_FEATURE_DISABLE

Номер бита	Условное обозначение	Назначение	Сброс
31:2	–	Резерв	?
1	DISABLE_EOMF_CHAR_REP	Замена последнего символа мультикадра. Когда включено, ядро заменит символ в последнем мультикадре на управляющий символ в соответствии со стандартом. При отключении ядро не заменяет последний символ в мультикадре. 0 – включено 1 – отключено	0
0	INITIAL_LANE_ALIGNMENT	Начальное выравнивание полос. Когда выравнивание включено, контроллер отправит последовательность ILAS после подтверждения sync_b. Если этот параметр отключен, входящие данные будут отправлены после синхронизации группы кодов. 0 – включено 1 – отключено	0

25.2.2.22 Регистр TX_CORE_INTERRUPT_MASK

Регистр маски запросов на прерывание.

0 – источник прерывания не маскируется

1 – источник прерывания маскируется

Формат регистра TX_CORE_INTERRUPT_MASK приведен в Таблица 25.139.

Таблица 25.139. Формат регистра TX_CORE_INTERRUPT_MASK

Номер бита	Условное обозначение	Назначение	Сброс
31:4	–	Резерв	0
3	Detected_over_run_in_TX_datapath	Маска прерывания по переполнению	1
2	Detected_under_run_in_TX_datapath	Маска прерывания по недозагруженности	1
1	Signaled_resynchronization_request_on_sync_b	Маска прерывания по запрос на повторную синхронизацию сигнала Sync_b.	1
0	Signaled_error_on_sync_b	Маска прерывания по ошибке синхронизации	1

25.2.2.23 Регистр TX_CORE_INTERRUPT_STATUS

Регистр состояния запросов прерывания. Аналогичен TX_CORE_INTERRUPT_MASK, за исключением того, что поля регистра имеют доступ типа RC, т.е. сбрасываются после чтения.

Формат регистра TX_CORE_INTERRUPT_STATUS приведен в Таблица 25.140.

Таблица 25.140. Формат регистра TX_CORE_INTERRUPT_STATUS

Номер бита	Условное обозначение	Назначение	Сброс
31:4	–	Резерв	0
3	Detected_over_run_in_TX_datapath	Запрос прерывания по переполнению	0
2	Detected_under_run_in_TX_datapath	Запрос прерывания по недозагруженности	0
1	Signaled_resynchronization_request_on_sync_b	Запрос прерывания по запрос на повторную синхронизацию сигнала Sync_b.	0
0	Signaled_error_on_sync_b	Запрос прерывания по ошибке синхронизации	0

25.2.2.24 Регистр TX TX_ILAS_AND_MISC_DETAILS

Регистр хранения информации о длине ILAS и другие параметры линков.

Формат регистра TX_ILAS_AND_MISC_DETAILS приведен в Таблица 25.141.

Таблица 25.141. Формат регистра TX_ILAS_AND_MISC_DETAILS

Номер бита	Условное обозначение	Назначение	Сброс
31:24	NUM_MULTIFRAMES_IN_ILAS	Количество мультикадров в ILAS. Допустимые значения: от 3 до 255	8'h3
23:18	Sync_b_lh_to_lmfc	Количество тактов слева от sync_b до следующей локальной границы многокадровых часов. Доступ – RO	0
17	PHADJ	Значение, которое должно быть передано как часть мультикадра 2 ILAS	0
16	ADJDIR	Значение, которое должно быть передано как часть мультикадра 2 ILAS	0
15:12	ADJCNT	Значение, которое должно быть передано как часть мультикадра 2 ILAS	0
11	–	Резерв	?
10:0	MULTIFRAME_LENGTH_OCTETS	Это поле определяет длину мультикадра в октетах	11'h11

25.2.2.25 Регистр PRBS_TEST_ENABLE

Регистр включения тестирования PRBS для индивидуальных линков

Формат регистра PRBS_TEST_ENABLE приведен в Таблица 25.142.

Таблица 25.142. Формат регистра PRBS_TEST_ENABLE

Номер бита	Условное обозначение	Назначение	Сброс
31:4	–	Резерв	?
3	PRBS_TEST_ENABLE_3	Включение PRBS теста для линка 3	0
2	PRBS_TEST_ENABLE_2	Включение PRBS теста для линка 2	0
1	PRBS_TEST_ENABLE_1	Включение PRBS теста для линка 1	0
0	PRBS_TEST_ENABLE_0	Включение PRBS теста для линка 0	0

25.2.2.26 Регистр JTSPAT_TEST_ENABLE

Регистр включения тестирования JTSPAT для всех линков

Формат регистра JTSPAT_TEST_ENABLE приведен в Таблица 25.143.

Таблица 25.143. Формат регистра JTSPAT_TEST_ENABLE

Номер бита	Условное обозначение	Назначение	Сброс
31:1	–	Резерв	?

Номер бита	Условное обозначение	Назначение	Сброс
0	JTSPAT_TEST_ENABLE	включение тестирования JTSPAT для всех линков	0

25.2.2.27 Регистр CLK_RATIO

Регистр определяющий соотношение между clk_sample и clk_character

Формат регистра CLK_RATIO приведен в Таблица 25.144.

Таблица 25.144. Формат регистра CLK_RATIO

Номер бита	Условное обозначение	Назначение	Сброс
31:2	–	Резерв	?
1:0	CLK_RATIO	Соотношение между clk_sample и clk_character «00» - clk_sample is = clk_character «01» - clk_sample is = 2* clk_character «10» - clk_sample is = 4* clk_character «11» - clk_sample is = 8* clk_character	0

25.2.2.28 Регистр TX_ILAS_DELAY

Регистр определяющий задержку между начальным запросом на TX интерфейсе и запросом канального уровень первого кадра из транспортного уровня. Единицы измерения – такты clk_character. Грубая формула:

- 4 – если F=1
- 6 – если F=2
- 8 – если F=3
- 8+(F/4) – если F>3

Для F> 3 значение может нуждаться в тонкой настройке, особенно если соотношение между clk_character и clk_sample не равно 1:1.

Данные рекомендации должны работать в большинстве случаев, однако, пользователю рекомендуется проверить настройку в симуляциях для конкретных случаев.

Формат регистра TX_ILAS_DELAY приведен в Таблица 25.145.

Таблица 25.145. Формат регистра TX_ILAS_DELAY

Номер бита	Условное Обозначение	Назначение	Сброс
31:7	–	Резерв	?
6:0	TX_ILAS_DELAY	Задержка от первоначальных запросов	4

25.2.2.29 Регистр SOFT_SYNC

Регистр управления программной синхронизацией

Формат регистра SOFT_SYNC приведен в Таблица 25.146.

Таблица 25.146. Формат регистра SOFT_SYNC

Номер бита	Условное обозначение	Назначение	Сброс
31:1	–	Резерв	?
0	Soft_Sync	Программная синхронизация 0 – синхронизация не используется 1 – синхронизация используется.	0

25.2.2.30 Регистр TL_TEST_MODE

Регистр включения и выбора режима тестирования транспортного уровня

Формат регистра TL_TEST_MODE приведен в Таблица 25.147.

Таблица 25.147. Формат регистра TL_TEST_MODE

Номер бита	Условное обозначение	Назначение	Сброс
31:2	–	Резерв	?
1	tl_test_pattern_mode	0 – выбран короткий тестовый шаблон 1 – выбран длинный тестовый шаблон	0
0	tl_test_mode_enable	0 – режим тестирования TL отключен 1 – режим тестирования TL включен	0

25.2.2.31 Регистр LMFC_DELAY

Регистр управления LMFC задержкой

Формат регистра LMFC_DELAY приведен в Таблица 25.148.

Таблица 25.148. Формат регистра LMFC_DELAY

Номер бита	Условное Обозначение	Назначение	Сброс
31:5	–	Резерв	?
4:0	tx_lmfc_delay	LMFC задержка	0

25.2.2.32 Регистр SYSREF_MAXSHIFT

Регистр управления сдвигом SYSREF

Формат регистра SYSREF_MAXSHIFT приведен в Таблица 25.149.

Таблица 25.149. Формат регистра SYSREF_MAXSHIFT

Номер бита	Условное Обозначение	Назначение	Сброс
31:5	–	Резерв	?
4:0	tx_sysref_max_shift_allowed	управление сдвигом SYSREF	0

25.2.3 Регистры блока JESD_RX

Перечень регистров приведен в Таблица 25.150

Таблица 25.150. Перечень программно-доступных регистров блока JESD_RX

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
BID	Bank ID	0x0	RW	0x000
DID	Device ID	0x0	RW	0x004
LID	Lane ID for lanes 0 to 3	0x18820	RW	0x008
SUBCLASS	SUBCLASS Control	0x1	RW	0x01C
CF	CF Control	0x0	RW	0x020
CS	CS Control	0x0	RW	0x024
F	F Control	0x0	RW	0x028
K	K Control	0x10	RW	0x02C
L	L Control	0x3	RW	0x030
N	N Control	0xF	RW	0x034
M	M Control	0x3	RW	0x038
NTOTAL	NTOTAL Control	0xF	RW	0x03C
S	S Control	0x0	RW	0x040
HD	HD Control	0x0	RW	0x044
SCR	SCR Control	0x0	RW	0x048
LINKCONFIGURATIONERROR	LINKCONFIGURATIONERRO R Control	0x0	RO	0x04C
ENABLEMODULE	ENABLEMODULE Control	0x0	RW	0x050
ENABLELINKTEST	ENABLELINKTEST Control	0x0	RW	0x054
LINKTEST_0_RECEIVECOUNTE R	LINKTEST_0_RECEIVECOUN TER Control	0x0	RO	0x058
LINKTEST_1_RECEIVECOUNTE R	LINKTEST_1_RECEIVECOUN TER Control	0x0	RO	0x05C
LINKTEST_2_RECEIVECOUNTE R	LINKTEST_2_RECEIVECOUN TER Control	0x0	RO	0x060
LINKTEST_3_RECEIVECOUNTE R	LINKTEST_3_RECEIVECOUN TER Control	0x0	RO	0x064
LINKTEST_0_ERRORCOUNTER	LINKTEST_0_ERRORCOUNTE R Control	0x0	RO	0x0C0
LINKTEST_1_ERRORCOUNTER	LINKTEST_1_ERRORCOUNTE R Control	0x0	RO	0x0C4
LINKTEST_2_ERRORCOUNTER	LINKTEST_2_ERRORCOUNTE R Control	0x0	RO	0x0C8
LINKTEST_3_ERRORCOUNTER	LINKTEST_3_ERRORCOUNTE R Control	0x0	RO	0x0CC
RX_FEATURE_DISABLE	RX_FEATURE_DISABLE Control	0x0	RW	0x120
RX_CORE_INTERRUPT_MASK	RX_CORE_INTERRUPT_MAS K Control	0x1FF	RW	0x124

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
RX_CORE_INTERRUPT_STATUS	RX_CORE_INTERRUPT_STATUS Control	0x0	RO	0x128
RX_ILAS_ADN_MISC_DETAILS	RX_ILAS_ADN_MISC_DETAILS Control	0x3011011	RW	0x12C
RX_ERR_INJ_REQ	RX_ERR_INJ_REQ Control	0x0	WO	0x130
PRBS_TEST_ENABLE	PRBS_TEST_ENABLE Control	0x0	RW	0x134
JTSPAT_TEST_ENABLE	JTSPAT_TEST_ENABLE Control	0x0	RW	0x138
CLK_RATIO	CLK_RATIO Control	0x0	RW	0x13C
PRBSTEST_0_ERRORCOUNTER	PRBSTEST_0_ERRORCOUNTER Control	0x0	RO	0x140
PRBSTEST_1_ERRORCOUNTER	PRBSTEST_1_ERRORCOUNTER Control	0x0	RO	0x144
PRBSTEST_2_ERRORCOUNTER	PRBSTEST_2_ERRORCOUNTER Control	0x0	RO	0x148
PRBSTEST_3_ERRORCOUNTER	PRBSTEST_3_ERRORCOUNTER Control	0x0	RO	0x148
JTSPATTEST_0_ERRORCOUNTER	JTSPATTEST_0_ERRORCOUNTER Control	0x0	RO	0x1A0
JTSPATTEST_1_ERRORCOUNTER	JTSPATTEST_1_ERRORCOUNTER Control	0x0	RO	0x1A4
JTSPATTEST_2_ERRORCOUNTER	JTSPATTEST_2_ERRORCOUNTER Control	0x0	RO	0x1A8
JTSPATTEST_3_ERRORCOUNTER	JTSPATTEST_3_ERRORCOUNTER Control	0x0	RO	0x1AC
PRBSTEST_0_RECEIVECOUNTER	PRBSTEST_0_RECEIVECOUNTER Control	0x0	RO	0x200
PRBSTEST_1_RECEIVECOUNTER	PRBSTEST_1_RECEIVECOUNTER Control	0x0	RO	0x204
PRBSTEST_2_RECEIVECOUNTER	PRBSTEST_2_RECEIVECOUNTER Control	0x0	RO	0x208
PRBSTEST_3_RECEIVECOUNTER	PRBSTEST_3_RECEIVECOUNTER Control	0x0	RO	0x20C
JTSPATTEST_0_RECEIVECOUNTER	JTSPATTEST_0_RECEIVECOUNTER Control	0x0	RO	0x260
JTSPATTEST_1_RECEIVECOUNTER	JTSPATTEST_1_RECEIVECOUNTER Control	0x0	RO	0x264
JTSPATTEST_2_RECEIVECOUNTER	JTSPATTEST_2_RECEIVECOUNTER Control	0x0	RO	0x268
JTSPATTEST_3_RECEIVECOUNTER	JTSPATTEST_3_RECEIVECOUNTER Control	0x0	RO	0x26C
SOFT_SYNC	SOFT_SYNC control	0x0	RW	0x2D0
TL_TEST_MODE	TL_TEST_MODE control	0x0	RW	0x2DC
RX_LMFC_DELAY	RX_LMFC_DELAY control	0x0	RW	0x2E0
RX_SYSREF_MAXSHIFT	RX_SYSREF_MAXSHIFT control	0x0	RW	0x2E4

25.2.3.1 Регистр BID

Формат регистра BID приведен в Таблица 25.151.

Таблица 25.151. Формат регистра BID

Номер бита	Условное обозначение	Назначение	Сброс
31:4	–	Резерв	?
3:0	BID	Bank ID	0

25.2.3.2 Регистр DID

Формат регистра DID приведен в Таблица 25.152.

Таблица 25.152. Формат регистра DID

Номер бита	Условное обозначение	Назначение	Сброс
31:8	–	Резерв	?
7:0	DID	Device ID	0

25.2.3.3 Регистр LID

Lane ID для полос от 0 до 3.

Определяет соотношение физической и логической полос. Например, если ‘h01 расположен в первых пяти битах [4: 0], соответствующих LID0, это означает, что на выходе канального уровня будет установлен вход с линии 1 в шине serdes_data_i на выход линии 0 в шине data_to_tl.

Формат регистра LID приведен в Таблица 25.153.

Таблица 25.153. Формат регистра LID

Номер бита	Условное обозначение	Назначение	Сброс
31:20	–	Резерв	?
19:15	LID3	Определяет соотношение физической и логической полос.	3
14:10	LID2	Определяет соотношение физической и логической полос.	2
9:5	LID1	Определяет соотношение физической и логической полос.	1
4:0	LID0	Определяет соотношение физической и логической полос.	0

25.2.3.4 Регистр SUBCLASS

Выбор режима работы – Subclass 0, 1, или 2.

Формат регистра SUBCLASS приведен в Таблица 25.154.

Таблица 25.154. Формат регистра SUBCLASS

Номер бита	Условное обозначение	Назначение	Сброс
31:3	–	Резерв	?
2:0	SUBCLASS	Выбор режима работы – Subclass 0, 1, или 2.	1

25.2.3.5 Регистр CF

Формат регистра CF приведен в Таблица 25.155.

Таблица 25.155. Формат регистра CF

Номер бита	Условное обозначение	Назначение	Сброс
31:18	–	Резерв	?
17:10	CF_FIRST_NIBBLE	Если CF = 1, это указывает на первую 4-битную позицию полубайта контрольных бит CF-lane	0
9:5	CF_FIRST_LANE	Указывает полосу, когда контрольные слова найдены	0
4:2	–	Резерв	?
1:0	CF	Количество контрольных слов на конвертер. (CF=0, or 1)	0

25.2.3.6 Регистр CS

Количество контрольных бит на семпл.

Формат регистра CS приведен в Таблица 25.156.

Таблица 25.156. Формат регистра CS

Номер бита	Условное обозначение	Назначение	Сброс
31:2	–	Резерв	?
1:0	CS	Количество контрольных бит на семпл	0

25.2.3.7 Регистр F

Количество октетов в кадре. Допустимые значения: 0-7. Кодировка - двоичное значение минус 1.

Формат регистра F приведен в Таблица 25.157.

Таблица 25.157. Формат регистра F

Номер бита	Условное обозначение	Назначение	Сброс
31:8	–	Резерв	?
7:0	F	Количество октетов в кадре	0

25.2.3.8 Регистр K

Количество кадров в мультикадре. Допустимые значения: 0-31. Кодировка - двоичное значение минус 1

Формат регистра K приведен в Таблица 25.158.

Таблица 25.158. Формат регистра K

Номер бита	Условное обозначение	Назначение	Сброс
31:5	–	Резерв	?
4:0	K	Количество кадров в мультикадре.	5'h10

25.2.3.9 Регистр L

Количество полос на конвертер. Допустимые значения: 0-3. Кодировка - двоичное значение минус 1.

Формат регистра L приведен в Таблица 25.159.

Таблица 25.159. Формат регистра L

Номер бита	Условное обозначение	Назначение	Сброс
31:5	–	Резерв	?
4:0	L	Количество полос на конвертер	5'h3

25.2.3.10 Регистр N

Разрешение конвертера. Допустимые значения: 7, 9, 11, 13, 14 и 15. Кодировка - двоичные значения минус 1.

Формат регистра N приведен в Таблица 25.160.

Таблица 25.160. Формат регистра N

Номер бита	Условное обозначение	Назначение	Сброс
31:5	–	Резерв	?
4:0	N	Разрешение конвертера	5'hF

25.2.3.11 Регистр M

Количество конвертеров на устройство. Допустимые значения: 0-7. Кодировка - двоичное значение минус 1.

Формат регистра M приведен в Таблица 25.161.

Таблица 25.161. Формат регистра M

Номер бита	Условное обозначение	Назначение	Сброс
31:8	–	Резерв	?

Номер бита	Условное обозначение	Назначение	Сброс
7:0	M	Количество конвертеров на устройство	8'h3

25.2.3.12 Регистр NTOTAL

Общее количество бит на семпл (N'). Должно быть целым числом, кратным 4. Допустимые значения: 7, 11, 15. Кодировка - двоичное значение минус 1.

Формат регистра NTOTAL приведен в Таблица 25.162.

Таблица 25.162. Формат регистра NTOTAL

Номер бита	Условное обозначение	Назначение	Сброс
31:5	–	Резерв	?
4:0	NTOTAL	Общее количество бит на семпл (N')	5'hF

25.2.3.13 Регистр S

Количество семплов на конвертер на цикл кадра. Допустимые значения: от 0 до 3. Кодировка - двоичное значение минус 1.

Формат регистра S приведен в Таблица 25.163.

Таблица 25.163. Формат регистра S

Номер бита	Условное обозначение	Назначение	Сброс
31:5	–	Резерв	?
4:0	S	Количество семплов на конвертер на цикл кадра	0

25.2.3.14 Регистр HD

Формат высокой плотности (HD).

Формат регистра HD приведен в Таблица 25.164.

Таблица 25.164. Формат регистра HD

Номер бита	Условное обозначение	Назначение	Сброс
31:1	–	Резерв	?
0	HD	Формат высокой плотности (HD) 0 – HD не используется 1 – HD используется.	0

25.2.3.15 Регистр SCR

Регистр управления скремблированием

Формат регистра SCR приведен в Таблица 25.165.

Таблица 25.165. Формат регистра SCR

Номер бита	Условное обозначение	Назначение	Сброс
31:1	–	Резерв	?
0	SCR	Включение скремблирования	0

25.2.3.16 Регистр LINKCONFIGURATIONERROR

Ошибка конфигурации линка.

Формат регистра LINKCONFIGURATIONERROR приведен в Таблица 25.166.

Таблица 25.166. Формат регистра LINKCONFIGURATIONERROR

Номер бита	Условное обозначение	Назначение	Сброс
31:4	–	Резерв	?
3:0	LINKCONFIGURATIONERROR	Указание на то, что данные, полученные во время последовательности синхронизации полосы, соответствуют конфигурации модуля Rx. 0 – нет несоответствия 1 – несоответствие	0

25.2.3.17 Регистр ENABLEMODULE

Регистр включения работы модуля

Формат регистра ENABLEMODULE приведен в Таблица 25.167.

Таблица 25.167. Формат регистра ENABLEMODULE

Номер бита	Условное обозначение	Назначение	Сброс
31:1	–	Резерв	?
0	ENABLEMODULE	0 – модуль отключен 1 – модуль включен.	0

25.2.3.18 Регистр ENABLELINKTEST

Регистр включения тестирование канального уровня.

Формат регистра ENABLELINKTEST приведен в Таблица 25.168.

Таблица 25.168. Формат регистра ENABLELINKTEST

Номер бита	Условное обозначение	Назначение	Сброс
31:2	–	Резерв	?

Номер бита	Условное обозначение	Назначение	Сброс
1:0	ENABLELINKTEST	0 – тест на канальном уровне не выполнен. 1 – передача непрерывной последовательности символов /K28.5/ (LINKTEST1). 2 – передача последовательности кодовой группы с последующей повторной последовательностью выравнивания дорожек (LINKTEST2). 3 – тест на канальном уровне не выполнен	0

25.2.3.19 Регистр LINKTEST_0_RECEIVECOUNTER

Количество полученных кадров или символов. Если включено тестирование канального уровня, то количество полученных символов (LINKTEST1) или кадров (LINKTEST2) считается для полосы 0 (lane 0)

Формат регистра LINKTEST_0_RECEIVECOUNTER приведен в Таблица 25.169.

Таблица 25.169. Формат регистра LINKTEST_0_RECEIVECOUNTER

Номер бита	Условное обозначение	Назначение	Сброс
31:0	LINKTEST_0_RECEIVECOUNTER	Количество полученных кадров или символов.	0

25.2.3.20 Регистр LINKTEST_1_RECEIVECOUNTER

Количество полученных кадров или символов. Если включено тестирование канального уровня, то количество полученных символов (LINKTEST1) или кадров (LINKTEST2) считается для полосы 1 (lane 1).

Формат регистра LINKTEST_1_RECEIVECOUNTER приведен в Таблица 25.170.

Таблица 25.170. Формат регистра LINKTEST_1_RECEIVECOUNTER

Номер бита	Условное Обозначение	Назначение	Сброс
31:0	LINKTEST_1_RECEIVECOUNTER	Количество полученных кадров или символов.	0

25.2.3.21 Регистр LINKTEST_2_RECEIVECOUNTER

Количество полученных кадров или символов. Если включено тестирование канального уровня, то количество полученных символов (LINKTEST1) или кадров (LINKTEST2) считается для полосы 2 (lane 2).

Формат регистра LINKTEST_2_RECEIVECOUNTER приведен в Таблица 25.171.

Таблица 25.171. Формат регистра LINKTEST_2_RECEIVECOUNTER

Номер бита	Условное Обозначение	Назначение	Сброс
31:0	LINKTEST_2_RECEIVECOUNTER	Количество полученных кадров или символов.	0

25.2.3.22 Регистр LINKTEST_3_RECEIVECOUNTER

Количество полученных кадров или символов. Если включено тестирование канального уровня, то количество полученных символов (LINKTEST1) или кадров (LINKTEST2) считается для полосы 3 (lane 3).

Формат регистра LINKTEST_3_RECEIVECOUNTER приведен в Таблица 25.172.

Таблица 25.172. Формат регистра LINKTEST_3_RECEIVECOUNTER

Номер бита	Условное Обозначение	Назначение	Сброс
31:0	LINKTEST_3_RECEIVECOUNTER	Количество полученных кадров или символов.	0

25.2.3.23 Регистр LINKTEST_0_ERRORCOUNTER

Счетчик количества ошибок в процессе канального тестирования для полосы 0 (lane 0).

Формат регистра LINKTEST_0_ERRORCOUNTER приведен в Таблица 25.173.

Таблица 25.173. Формат регистра LINKTEST_0_ERRORCOUNTER

Номер бита	Условное Обозначение	Назначение	Сброс
31:0	LINKTEST_0_ERRORCOUNTER	Количество обнаруженных ошибок	0

25.2.3.24 Регистр LINKTEST_1_ERRORCOUNTER

Счетчик количества ошибок в процессе канального тестирования для полосы 1 (lane 1).

Формат регистра LINKTEST_1_ERRORCOUNTER приведен в Таблица 25.174.

Таблица 25.174. Формат регистра LINKTEST_1_ERRORCOUNTER

Номер бита	Условное обозначение	Назначение	Сброс
31:0	LINKTEST_1_ERRORCOUNTER	Количество обнаруженных ошибок	0

25.2.3.25 Регистр LINKTEST_2_ERRORCOUNTER

Счетчик количества ошибок в процессе канального тестирования для полосы 2 (lane 2).

Формат регистра LINKTEST_2_ERRORCOUNTER приведен в Таблица 25.175.

Таблица 25.175. Формат регистра LINKTEST_2_ERRORCOUNTER

Номер бита	Условное обозначение	Назначение	Сброс
31:0	LINKTEST_2_ERRORCOUNTER	Количество обнаруженных ошибок	0

25.2.3.26 Регистр LINKTEST_3_ERRORCOUNTER

Счетчик количества ошибок в процессе канального тестирования для полосы 3 (lane 3).

Формат регистра LINKTEST_3_ERRORCOUNTER приведен в Таблица 25.176.

Таблица 25.176. Формат регистра LINKTEST_3_ERRORCOUNTER

Номер бита	Условное обозначение	Назначение	Сброс
31:0	LINKTEST_3_ERRORCOUNTER	Количество обнаруженных ошибок	0

25.2.3.27 Регистр RX_FEATURE_DISABLE

Регистр отключения функций модуля Rx для совместимости с другими устройствами.

Формат регистра RX_FEATURE_DISABLE приведен в Таблица 25.177.

Таблица 25.177. Формат регистра RX_FEATURE_DISABLE

Номер бита	Условное обозначение	Назначение	Сброс
31:5	–	Резерв	?
4	RE_INITIALIZATION_VIA_DATA_INTERFACE	Повторная инициализация через интерфейс данных. Когда эта опция включена, любая символьная ошибка приведет к повторной синхронизации. При отключении ошибки типа «not-in-table» и «disparity» не приведут к повторной синхронизации 0 – включено 1 – отключено	0
3	LANE_ALIGNMENT_PROCESSING	Обработка начальной последовательности выравнивания линий и замены символов. Когда этот параметр включен, регистр LINKCONFIGURATIONERROR будет показывать, соответствует ли полученный ILAS ожидаемой последовательности ILAS. При отключении никакая информация не будет отображаться в регистре LINKCONFIGURATIONERROR.	0

Номер бита	Условное обозначение	Назначение	Сброс
2	LANE_ALIGNMENT_CHARACTER_REPLACEMENT	Выравнивание линий и замена символов. Когда включено, контроллер заменит управляющие символы для соответствующих данных в соответствии со стандартом. Если отключено, управляющие символы останутся в выходных данных. 0 – включено 1 – отключено	0
1	LANE_ALIGNMENT_MONITORING	Контроль и коррекция выравнивания полос. Когда включено, ядро будет контролировать правильную отправку управляющих символов в ILAS. 0 – включено 1 – отключено	0
0	INITIAL_LANE_ALIGNMENT	Начальное выравнивание полос. Если включено, контроллер будет проверять наличие ILAS после того, как 4 контрольных символа k28.5 поступят в приемник. Если этот параметр отключен, входящие данные будут считаться стандартными данными после синхронизации группы кодов. 0 – включено 1 – отключено	0

25.2.3.28 Регистр RX_CORE_INTERRUPT_MASK

Регистр маски запросов на прерывание.

0 – источник прерывания не маскируется

1 – источник прерывания маскируется

Формат регистра RX_CORE_INTERRUPT_MASK приведен в Таблица 25.178.

Таблица 25.178. Формат регистра RX_CORE_INTERRUPT_MASK

Номер бита	Условное обозначение	Назначение	Сброс
31:9	–	Резерв	0
8	Code_group_synchronization_error	Маска прерывания по ошибке кодовой синхронизации (code group synchronization)	1
7	A_control_character_is_received_that_is_not_expected_at_the_given_character_position	Маска прерывания при обнаружении управляющих символов во время получения пользовательских данных /ILAS	1

Номер бита	Условное обозначение	Назначение	Сброс
6	Not_in_table_error	Маска прерывания при обнаружении «not in table error» во время получения пользовательских данных /ILAS	1
5	Unexpected_control_character	Маска прерывания при обнаружении «unexpected control character» во время получения пользовательских данных /ILAS	1
4	Detected_rx_disparity_errror	Маска прерывания при обнаружении «rx disparity error» во время получения пользовательских данных /ILAS	1
3	Detected_over_run_in_RX_data_path	Маска прерывания по переполнению внутренних модулей	1
2	Detected_under_run_in_RX_data_path	Маска прерывания по недозаполнению внутренних модулей	1
1	Signaled_resynchronization_request_on_sync_b	Маска прерывания по запросу на повторную синхронизацию по сигналу синхронизации.	1
0	Signaled_error_on_sync_b	Маска прерывания по ошибке в сигнале синхронизации.	1

25.2.3.29 Регистр RX_CORE_INTERRUPT_STATUS

Регистр состояния запросов прерывания. Аналогичен RX_CORE_INTERRUPT_MASK, за исключением того, что поля регистра имеют доступ типа RC, т.е. сбрасываются после чтения.

Формат регистра RX_CORE_INTERRUPT_STATUS приведен в Таблица 25.179.

Таблица 25.179. Формат регистра RX_CORE_INTERRUPT_STATUS

Номер бита	Условное обозначение	Назначение	Сброс
31:9	–	Резерв	0
8	Code_group_synchronization_error	Статус прерывания по ошибке кодовой синхронизации (code group synchronization)	0
7	A_control_character_is_received_that_is_not_ expected_at_the_given_character_position	Статус прерывания при обнаружении управляющих символов во время получения пользовательских данных /ILAS	0

Номер бита	Условное обозначение	Назначение	Сброс
6	Not_in_table_error	Статус прерывания при обнаружении «not in table error» во время получения пользовательских данных /ILAS	0
5	Unexpected_control_character	Статус прерывания при обнаружении «unexpected control character» во время получения пользовательских данных /ILAS	0
4	Detected_rx_disparity_errror	Статус прерывания при обнаружении «rx disparity error» во время получения пользовательских данных /ILAS	0
3	Detected_over_run_in_RX_data_path	Статус прерывания по переполнению внутренних модулей	0
2	Detected_under_run_in_RX_data_path	Статус прерывания по недозаполнению внутренних модулей	0
1	Signaled_resynchronization_request_on_sync_b	Статус прерывания по запросу на повторную синхронизацию по сигналу синхронизации.	0
0	Signaled_error_on_sync_b	Статус прерывания по ошибке в сигнале синхронизации.	0

25.2.3.30 Регистр RX_ILAS_ADN_MISC_DETAILS

Регистр хранения информации о длине ILAS и RBD.

Формат регистра RX_ILAS_ADN_MISC_DETAILS приведен в Таблица 25.180.

Таблица 25.180. Формат регистра RX_ILAS_ADN_MISC_DETAILS

Номер бита	Условное обозначение	Назначение	Сброс
31:24	NUM_MULTIFRAMES_IN_ILAS	Количество мультикадров в ILAS. Допустимые значения: от 3 до 255. Фактическая длина - это запрограммированное значение + 1	8'h3
23:12	RBD_LENGTH_OCTETS	Длина кадров RBD в октетах.	12'h11
11:0	MULTIFRAME_LENGTH_OCTETS	Длину мультикадра в октетах (F*K).	12'h11

25.2.3.31 Регистр RX_ERR_INJ_REQ

Регистр внесения ошибок для проверки совместимости с прошивкой.

Формат регистра RX_ERR_INJ_REQ приведен в Таблица 25.181.

Таблица 25.181. Формат регистра RX_ERR_INJ_REQ

Номер бита	Условное обозначение	Назначение	Сброс
31:24	Inject_frame_alignment_error	Запись 1 в этот бит приведет к замене 2 последовательных символов на serdes интерфейсе на / F /	0
23:16	Inject_Lane_alignment_error	Запись 1 в этот бит приведет к замене 2 последовательных символов на интерфейсе serdes на / A /	0
15:8	Inject_not_in_table_error	Запись 1 в этот бит приведет к тому, что 1 символ на интерфейсе serdes не будет отображаться в таблице ошибок	0
7:0	Req_Resynchronization	Запись 1 в этот бит приведет к тому, что 4 последовательных символа на интерфейсе serdes не будут отображаться в таблице ошибок	0

25.2.3.32 Регистр PRBS_TEST_ENABLE

Регистр включения тестирования PRBS для индивидуальных линков.

Формат регистра PRBS_TEST_ENABLE приведен в Таблица 25.182.

Таблица 25.182. Формат регистра PRBS_TEST_ENABLE

Номер бита	Условное обозначение	Назначение	Сброс
31:4	–	Резерв	?
3	PRBS_TEST_ENABLE_3	Включение PRBS теста для линка 3	0
2	PRBS_TEST_ENABLE_2	Включение PRBS теста для линка 2	0
1	PRBS_TEST_ENABLE_1	Включение PRBS теста для линка 1	0
0	PRBS_TEST_ENABLE_0	Включение PRBS теста для линка 0	0

25.2.3.33 Регистр JTSPAT_TEST_ENABLE

Регистр включения тестирования JTSPAT для индивидуальных линков.

Формат регистра JTSPAT_TEST_ENABLE приведен в Таблица 25.183.

Таблица 25.183. Формат регистра JTSPAT_TEST_ENABLE

Номер бита	Условное обозначение	Назначение	Сброс
31:4	–	Резерв	?
3	JTSPAT_TEST_ENABLE_3	Включение JTSPAT теста для линка 3	0
2	JTSPAT_TEST_ENABLE_2	Включение JTSPAT теста для линка 2	0
1	JTSPAT_TEST_ENABLE_1	Включение JTSPAT теста для линка 1	0
0	JTSPAT_TEST_ENABLE_0	Включение JTSPAT теста для линка 0	0

25.2.3.34 Регистр CLK_RATIO

Регистр определяющий соотношение между clk_sample и clk_character.

Формат регистра CLK_RATIO приведен в Таблица 25.184.

Таблица 25.184. Формат регистра CLK_RATIO

Номер бита	Условное обозначение	Назначение	Сброс
31:2	–	Резерв	?
1:0	CLK_RATIO	Соотношение между clk_sample и clk_character «00» - clk_sample is = clk_character «01» - clk_sample is = 2* clk_character «10» - clk_sample is = 4* clk_character «11» - clk_sample is = 8* clk_character	0

25.2.3.35 Регистр PRBSTEST_0_ERRORCOUNTER

Счетчик обнаруженных ошибок в процессе PRBS тестирования для линка 0 (lane 0).

Формат регистра PRBSTEST_0_ERRORCOUNTER приведен в Таблица 25.185.

Таблица 25.185. Формат регистра PRBSTEST_0_ERRORCOUNTER

Номер бита	Условное обозначение	Назначение	Сброс
31:0	PRBSTEST_0_ERRORCOUNTER	Количество обнаруженных ошибок	0

25.2.3.36 Регистр PRBSTEST_1_ERRORCOUNTER

Счетчик обнаруженных ошибок в процессе PRBS тестирования для линка 1 (lane 1).

Формат регистра PRBSTEST_1_ERRORCOUNTER приведен в Таблица 25.186.

Таблица 25.186. Формат регистра PRBSTEST_1_ERRORCOUNTER

Номер бита	Условное обозначение	Назначение	Сброс
31:0	PRBSTEST_1_ERRORCOUNTER	Количество обнаруженных ошибок	0

25.2.3.37 Регистр PRBSTEST_2_ERRORCOUNTER

Счетчик обнаруженных ошибок в процессе PRBS тестирования для линка 2 (lane 2).

Формат регистра PRBSTEST_2_ERRORCOUNTER приведен в Таблица 25.187.

Таблица 25.187. Формат регистра PRBSTEST_2_ERRORCOUNTER

Номер бита	Условное обозначение	Назначение	Сброс
31:0	PRBSTEST_2_ERRORCOUNTER	Количество обнаруженных ошибок	0

25.2.3.38 Регистр PRBSTEST_3_ERRORCOUNTER

Счетчик обнаруженных ошибок в процессе PRBS тестирования для линка 3 (lane 3).

Формат регистра PRBSTEST_3_ERRORCOUNTER приведен в Таблица 25.188.

Таблица 25.188. Формат регистра PRBSTEST_3_ERRORCOUNTER

Номер бита	Условное обозначение	Назначение	Сброс
31:0	PRBSTEST_3_ERRORCOUNTER	Количество обнаруженных ошибок	0

25.2.3.39 Регистр JTSPATTEST_0_ERRORCOUNTER

Счетчик обнаруженных ошибок в процессе JTSPAT тестирования для линка 0 (lane 0).

Формат регистра JTSPATTEST_0_ERRORCOUNTER приведен в Таблица 25.189.

Таблица 25.189. Формат регистра JTSPATTEST_0_ERRORCOUNTER

Номер бита	Условное обозначение	Назначение	Сброс
31:0	JTSPATTEST_0_ERRORCOUNTER	Количество обнаруженных ошибок	0

25.2.3.40 Регистр JTSPATTEST_1_ERRORCOUNTER

Счетчик обнаруженных ошибок в процессе JTSPAT тестирования для линка 1 (lane 1).

Формат регистра JTSPATTEST_1_ERRORCOUNTER приведен в Таблица 25.190.

Таблица 25.190. Формат регистра JTSPATTEST_1_ERRORCOUNTER

Номер бита	Условное обозначение	Назначение	Сброс
31:0	JTSPATTEST_1_ERRORCOUNTER	Количество обнаруженных ошибок	0

25.2.3.41 Регистр JTSPATTEST_2_ERRORCOUNTER

Счетчик обнаруженных ошибок в процессе JTSPAT тестирования для линка 2 (lane 2).

Формат регистра JTSPATTEST_2_ERRORCOUNTER приведен в Таблица 25.191.

Таблица 25.191. Формат регистра JTSPATTEST_2_ERRORCOUNTER

Номер бита	Условное обозначение	Назначение	Сброс
31:0	JTSPATTEST_2_ERRORCOUNTER	Количество обнаруженных ошибок	0

25.2.3.42 Регистр JTSPATTEST_3_ERRORCOUNTER

Счетчик обнаруженных ошибок в процессе JTSPAT тестирования для линка 3 (lane 3).

Формат регистра JTSPATTEST_3_ERRORCOUNTER приведен в Таблица 25.192.

Таблица 25.192. Формат регистра JTSPATTEST_3_ERRORCOUNTER

Номер бита	Условное обозначение	Назначение	Сброс
31:0	JTSPATTEST_3_ERRORCOUNTER	Количество обнаруженных ошибок	0

25.2.3.43 Регистр PRBSTEST_0_RECEIVECOUNTER

Счетчик полученных последовательностей PRBS в процессе PRBS тестирования для линка 0 (lane 0).

Формат регистра PRBSTEST_0_RECEIVECOUNTER приведен в Таблица 25.193.

Таблица 25.193. Формат регистра PRBSTEST_0_RECEIVECOUNTER

Номер бита	Условное обозначение	Назначение	Сброс
31:0	PRBSTEST_0_RECEIVECOUNTER	Количество полученных последовательностей prbs	0

25.2.3.44 Регистр PRBSTEST_1_RECEIVECOUNTER

Счетчик полученных последовательностей PRBS в процессе PRBS тестирования для линка 1 (lane 1).

Формат регистра PRBSTEST_1_RECEIVECOUNTER приведен в Таблица 25.194.

Таблица 25.194. Формат регистра PRBSTEST_1_RECEIVECOUNTER

Номер бита	Условное обозначение	Назначение	Сброс
31:0	PRBSTEST_1_RECEIVECOUNTER	Количество полученных последовательностей prbs	0

25.2.3.45 Регистр PRBSTEST_2_RECEIVECOUNTER

Счетчик полученных последовательностей PRBS в процессе PRBS тестирования для линка 2 (lane 2).

Формат регистра PRBSTEST_2_RECEIVECOUNTER приведен в Таблица 25.195.

Таблица 25.195. Формат регистра PRBSTEST_2_RECEIVECOUNTER

Номер бита	Условное обозначение	Назначение	Сброс
31:0	PRBSTEST_2_RECEIVECOUNTER	Количество полученных последовательностей prbs	0

25.2.3.46 Регистр PRBSTEST_3_RECEIVECOUNTER

Счетчик полученных последовательностей PRBS в процессе PRBS тестирования для линка 3 (lane 3).

Формат регистра PRBSTEST_3_RECEIVECOUNTER приведен в Таблица 25.196.

Таблица 25.196. Формат регистра PRBSTEST_3_RECEIVECOUNTER

Номер бита	Условное обозначение	Назначение	Сброс
31:0	PRBSTEST_3_RECEIVECOUNTER	Количество полученных последовательностей prbs	0

25.2.3.47 Регистр JTSPATTEST_0_RECEIVECOUNTER

Счетчик полученных последовательностей JTSPAT в процессе JTSPAT тестирования для линка 0 (lane 0).

Формат регистра JTSPATTEST_0_RECEIVECOUNTER приведен в Таблица 25.197.

Таблица 25.197. Формат регистра JTSPATTEST_0_RECEIVECOUNTER

Номер бита	Условное обозначение	Назначение	Сброс
31:0	JTSPATTEST_0_RECEIVECOUNTER	Количество полученных JTSPAT последовательностей	0

25.2.3.48 Регистр JTSPATTEST_1_RECEIVECOUNTER

Счетчик полученных последовательностей JTSPAT в процессе JTSPAT тестирования для линка 1 (lane 1).

Формат регистра JTSPATTEST_1_RECEIVECOUNTER приведен в Таблица 25.198.

Таблица 25.198. Формат регистра JTSPATTEST_1_RECEIVECOUNTER

Номер бита	Условное обозначение	Назначение	Сброс
31:0	JTSPATTEST_1_RECEIVECOUNTER	Количество полученных JTSPAT последовательностей	0

25.2.3.49 Регистр JTSPATTEST_2_RECEIVECOUNTER

Счетчик полученных последовательностей JTSPAT в процессе JTSPAT тестирования для линка 2 (lane 2).

Формат регистра JTSPATTEST_2_RECEIVECOUNTER приведен в Таблица 25.199.

Таблица 25.199. Формат регистра JTSPATTEST_2_RECEIVECOUNTER

Номер бита	Условное обозначение	Назначение	Сброс
31:0	JTSPATTEST_2_RECEIVECOUNTER	Количество полученных JTSPAT последовательностей	0

25.2.3.50 Регистр JTSPATTEST_3_RECEIVECOUNTER

Счетчик полученных последовательностей JTSPAT в процессе JTSPAT тестирования для линка 3 (lane 3).

Формат регистра JTSPATTEST_3_RECEIVECOUNTER приведен в Таблица 25.200.

Таблица 25.200. Формат регистра JTSPATTEST_3_RECEIVECOUNTER

Номер бита	Условное обозначение	Назначение	Сброс
31:0	JTSPATTEST_3_RECEIVECOUNTER	Количество полученных JTSPAT последовательностей	0

25.2.3.51 Регистр SOFT_SYNC

Регистр управления программной синхронизацией.

Формат регистра SOFT_SYNC приведен в Таблица 25.201.

Таблица 25.201. Формат регистра SOFT_SYNC

Номер бита	Условное обозначение	Назначение	Сброс
31:1	–	Резерв	?
0	Soft_Sync	Программная синхронизация 0 – синхронизация не используется 1 – синхронизация используется.	0

25.2.3.52 Регистр TL_TEST_MODE

Регистр включения режима тестирования транспортного уровня и отображения ошибок.

Формат регистра TL_TEST_MODE приведен в Таблица 25.202.

Таблица 25.202. Формат регистра TL_TEST_MODE

Номер бита	Условное обозначение	Назначение	Сброс
31:3	–	Резерв	?
2	tl_test_mode_error_hold_capture	Признак наличия ошибки. Тип доступа – RC	0
1	tl_test_pattern_mode	0 – выбран короткий тестовый шаблон 1 – выбран длинный тестовый шаблон	0
0	tl_test_mode_enable	0 – режим тестирования TL отключен 1 – режим тестирования TL включен	0

25.2.3.53 Регистр RX_LMFC_DELAY

Регистр управления LMFC задержкой.

Формат регистра RX_LMFC_DELAY приведен в Таблица 25.203.

Таблица 25.203. Формат регистра RX_LMFC_DELAY

Номер бита	Условное обозначение	Назначение	Сброс
31:5	–	Резерв	?
4:0	rx_lmfc_delay	lmfc задержка	0

25.2.3.54 Регистр RX_SYSREF_MAXSHIFT

Регистр управления сдвигом SYSREF.

Формат регистра RX_SYSREF_MAXSHIFT приведен в Таблица 25.204.

Таблица 25.204. Формат регистра RX_SYSREF_MAXSHIFT

Номер бита	Условное Обозначение	Назначение	Сброс
31:5	–	Резерв	?

Номер бита	Условное Обозначение	Назначение	Сброс
4:0	rx_sysref_max_shift_allowed	управление сдвигом SYSREF	0

25.3Функционирование контроллера Jesd204b

25.3.1Приемник

Функция приемного модуля состоит в том, чтобы получать и преобразовывать данные полученные с 4 высокоскоростных сериализованных полос (lanes) через интерфейс Serdes и передавать их в пользовательский параллельный интерфейс семплов (интерфейс прикладного уровня).

Приемник также отвечает за формирование сигнала синхронизации Sync_b.

На Рисунок 25.4 представлена архитектура Rx модуля.

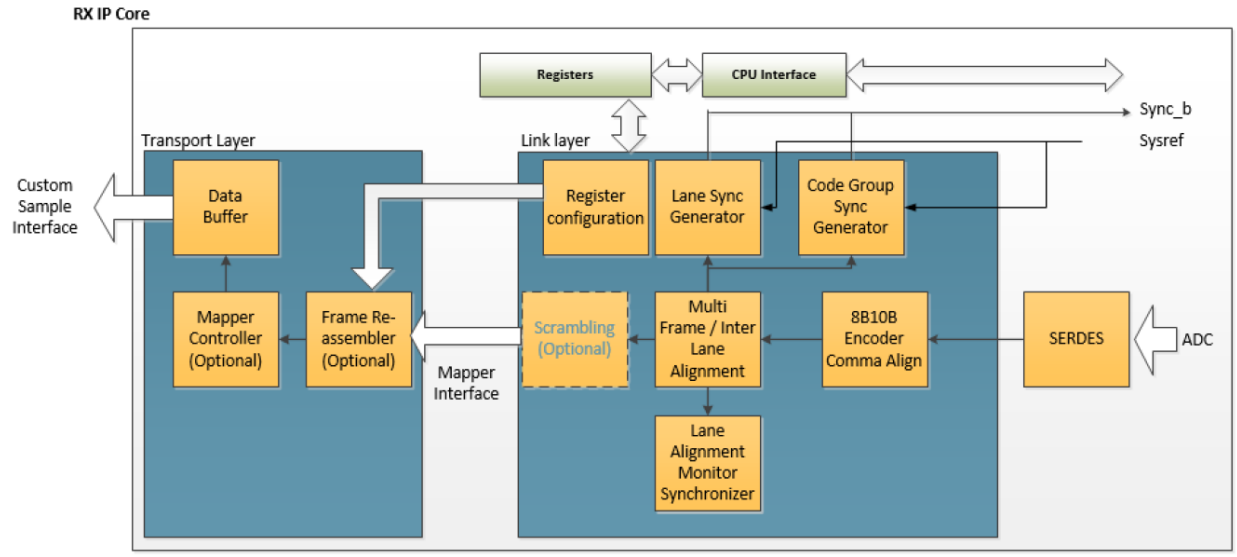


Рисунок 25.4. Архитектура Rx модуля

Модуль приемника разделен на два основных блока в соответствии со стандартом, канальный уровень (Link Layer) и транспортный уровень (Transport Layer).

Канальный уровень является основным ядром, тогда как транспортный уровень используется только для преобразования данных, полученных на канальном уровне.

Все блоки приемного модуля управляются регистрами, показанными в верхней части рисунка. Они взаимодействуют с ЦП для целей управления, конфигурирования и мониторинга, а поток данных направляется непосредственно в приложение.

25.3.2Передатчик

Функция передающего модуля состоит в том, чтобы получать данные из параллельного пользовательского интерфейса сэмплов, и передавать их на 4 высокоскоростных сериализованных полосы (lanes) Serdes интерфейса.

Передатчик также отвечает за прием и интерпретацию сигнала Sync_b, выдаваемого приемником, для целей синхронизации.

На Рисунок 25.5 представлена архитектура TX модуля.

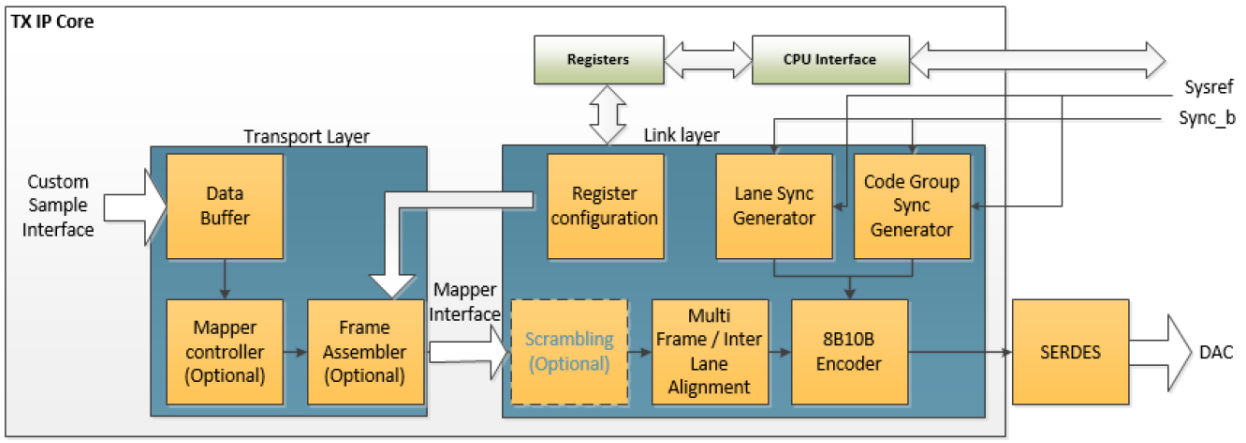


Рисунок 25.5. Архитектура TX модуля

Все блоки модуля передатчика управляются регистрами, показанными в верхней части рисунка. Они взаимодействуют с ЦП для целей управления, конфигурирования и мониторинга, а поток данных проходит через блоки к интерфейсу Serdes.

25.3.3 Интерфейс синхронизации

Sync_b

Используется в критичном по времени возврата пути от RX к TX, синхронно с clk_character. Этот сигнал используется для выдачи запроса на синхронизацию (инициирования процесса “Code Group Synchronization”) от RX к TX. Сигнал активен для всех подклассов и поддерживает детерминированную задержку в подклассе 2.

Sysref

В “Subclass 1” детерминированная задержка определяется с помощью sysref, этот сигнал распространяется на все устройства в системе, его целью является идентификация фронта тактового сигнала устройства (device clock), который будет использоваться для выравнивания фазы внутреннего LMFC и тактового сигнала кадра.

Sysref не используется ни подклассом 0, ни подклассом 2.

25.3.4 Частотные домены

Домены синхронизации представлены на Рисунок 25.6 – для Tx и на Рисунок 25.7 – для Rx модулей.

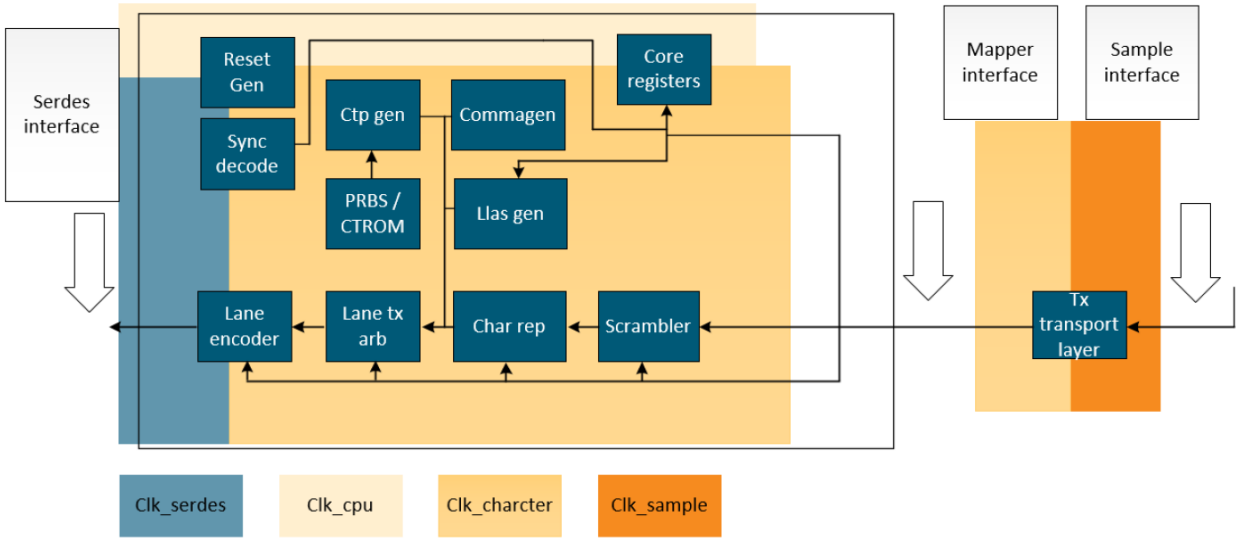


Рисунок 25.6. Домены синхронизации TX модуля

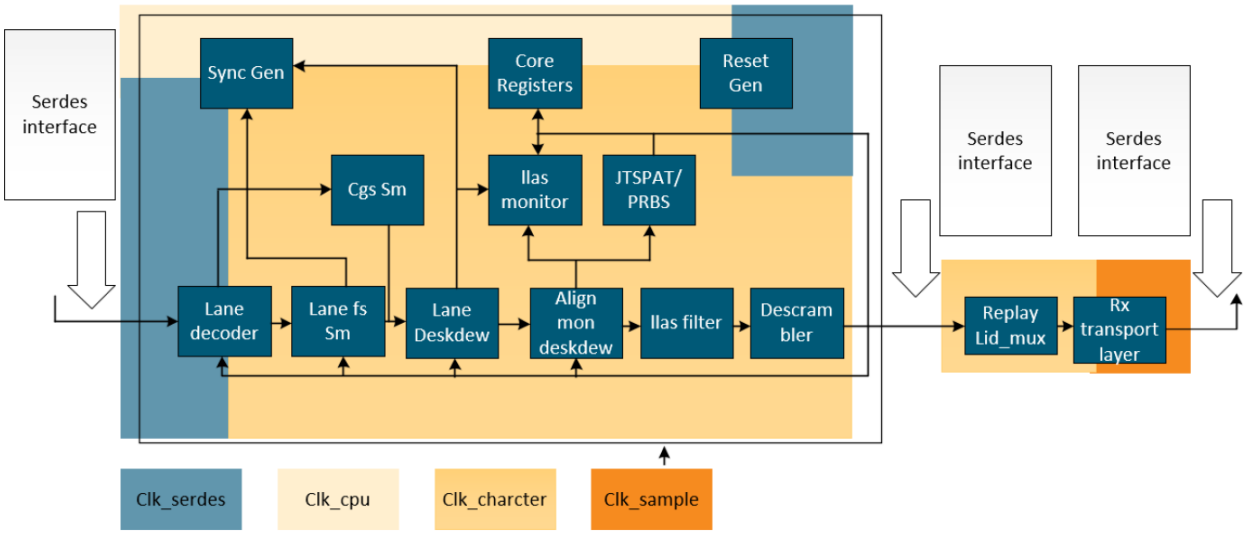


Рисунок 25.7. Домены синхронизации RX модуля

Расчет номинала частот начинается с выбора скорость линии. Например, для скорости линии 12,5 Гбит/с частота clk_serdes будет равна 625 МГц, частота clk_character равна 312,5 МГц.

Соотношение частот

- clk_character - базовая частота устройства (device clock). От этой частоты должны рассчитываться частоты clk_serdes и clk_sample
- clk_serdes - вдвое выше частоты clk_character
- clk_sample - может быть равна, либо быть выше в 2, 4, 8 раз частоты clk_character в зависимости от соотношения параметров F*L*8 и M*S*N

- clk_cpu – частота доступа к программным регистрам. Эта частота не зависит от частоты clk_character

Чтобы рассчитать соотношение между clk_sample и clk_character, необходимо знать соотношение между F, L с одной стороны и M, S и N' с другой. Соотношение между частотами clk_character и clk_sample должен быть 1: 1, 1: 2, 1: 4 или 1: 8.

Например, если F = 1 и L = 4, а S = 2, M = 2 и N = N '= 16:

F = 1 означает, что в каждом такте clk_character обрабатывается 4 кадра, поскольку каждая линия обрабатывает 32 бита параллельно.

L = 4 соответственно интерфейс семплов должен пропускать 4 (линии) * 4 (кадра) * 8 (бит) = 128 бит на такт clk_character.

При N = 16 и M = 2 интерфейс семплов может пропустить 32 бита за цикл. Таким образом, чтобы обеспечить 128 бит на такт clk_character, clk_sample должен иметь частоту 4 * clk_character.

Кроме этого, важно не забыть прописать данное соотношение в регистр CLK_RATIO.

25.3.5 Блок Replay_Lid_mux

В соответствии с пунктом 7.6.2 стандарта JESD, когда транспортный уровень используется для приема данных от канального уровня, кадр должен быть сохранен, чтобы в случае обнаружения в декодере ошибки "not-in-table error" последний кадр мог быть повторен для транспортного уровня. Блок Replay_lid_mux предназначен для хранения кадра, в соответствии стандартом JESD.

25.3.6 Поиск запятой

Средство поиска запятой ищет во входных данных запятую и выравнивает ее по 10-битным границам. Поиск запятой активируется при запуске и отключается после синхронизации ядра.

25.3.7 PRBS-15 генератор и анализатор

Чтобы выполнить тестирование тракта передачи-приема данных, в ядре реализованы генератор и анализатор PRBS-15. Они основаны на случайной схеме PRBS-15.

Анализатор может быть активирован для каждой полосы (lane), и количество действительных данных и ошибок будет показано в регистрах счетчиках PRBS действительных данных и PRBS ошибок соответственно.

Процедура тестирования PRBS:

- Дождитесь отправки ILAS и получения данных.
- Включить PRBS тестирование в TX в регистре «PRBS_TEST_ENABLE».
- Включите PRBS тестирование в RX в регистре «PRBS_TEST_ENABLE».
- Подождите, пока не будет отправлена вся последовательность PRBS (2 ^ 15 слов).
- Прочтите регистры «PRBSTEST_*_*COUNTER».

25.3.8 JTSPAT генератор и анализатор

Для контроля джиттера в контроллере реализованы генератор и анализатор JTSPAT. Они основаны на физическом интерфейсе “Fibre Channel-4” (FC-PI-4) в соответствии с американским национальным стандартом информационных технологий.

Генератор JTSPAT имеет память хранения шаблонов. Шаблон передается контроллером, когда включено тестирование JTSPAT.

Анализатор сохраняет всю последовательность из 118 8-битных слов, и активирует соответствующий флаг, если вся последовательность соответствует шаблону JTSPAT.

Количество достоверных и ошибочных последовательностей отображается в регистрах «JTSPATTEST_*_RECEIVECOUNTER» и «JTSPATTEST_*_ERRORCOUNTER».

25.3.9 Поддержка подклассов (Subclasses)

Контроллер JESD204B IP поддерживает подклассы 0, 1 и 2 (Subclasses 0, 1, 2). Subclass 0 не поддерживает детерминированную задержку. Детерминированная задержка Subclass 1 и Subclass 2 поддерживается через интерфейсы sysref и sync_b соответственно.

25.3.9.1 Subclass 0

Подкласс 0 разработан для поддержки обратной совместимости со стандартом JESD204A с работой на скоростях до 12,5 Гбит/с стандарта JESD204B. Это означает, что детерминированная задержка не поддерживается.

Модуль TX передает начальное выравнивание полосы (ILA) и пользовательские данные после подтверждения sync_b. После генерации ILA отсчеты будут запрашиваться из интерфейса семплов и передаваться в первом кадре после последовательности ILA.

RX обработает данные и доставит их в интерфейс семплов в момент, когда станут доступны данные в полосе с наибольшей задержкой.

25.3.9.2 Subclass 1

Для устройств подкласса 1 детерминированная задержка достигается с использованием sysref, который распространяется на все конвертеры и устройства. Рекомендуется, чтобы sysref генерировался из того же источника, что и синхросигнал device_clock.

После установки sysref приемник выравнивает внутренний многокадровый сигнал (LMFC) с сигналом sysref (с задержкой на 1 такт частоты clk_character), затем RX ожидает синхронизации группы кодов (K28.5) для снятия сигнала sync_b. После этого счетчик LMFC отслеживает количество октетов, переданных в каждом периоде мультикадра.

Снятие сигнала sync_b означает готовность RX к получению начальной последовательности выравнивания дорожек (ILAS) и пользовательских данных. После снятие сигнала sync_b передатчик начнет отправлять ILAS на локальной границе мультикадра (LMFC).

Приемник получит ILAS после следующей задержки:

- TX Delay: задержка от параллельной генерации ILA TX (которая выровнена по границе LMFC) до появления ILA на выходе SerDes TX.
- Lane Delay: задержка по внешнему физическому каналу.
- RX Delay: задержка от входа RX SerDes до вывода эластичного буфера. Начало ILA и/или начало пользовательских данных появятся на выходе эластичного буфера с выравниванием по границе LMFC + RBD.

В контроллере JESD204b учитываются следующие задержки (измеренные в тактовых тактах clk_character):

Таблица 25.205. Задержка в TX (LINK LAYER)

Блок	Частотный домен	Количество тактов
Scrambler	clk_character	1
Char rep	clk_character	2
Tx arbiter	clk_character	5
Lane Encoder	clk_character	4
Serializer	clk_serdes	Not included

Таблица 25.206. Задержка в RX (LINK LAYER)

Блок	Частотный домен	Количество тактов
Deserializer	clk_serdes	Not included
Lane Decoder	clk_character	5

Блок	Частотный домен	Количество тактов
FS_SM	clk_character	2
Deskew	clk_character	Up to LMFC
Align mon_char	clk_character	1
Ilas_filter	clk_character	4
Lid_mux	clk_character	3

На Рисунок 25.8 представлен процесс синхронизации модуля приемника и передатчика (для операции Subclass 1) и выравнивания данных.

Именно буфер приема RX хранит данные во всех дорожках до следующего LMFC.

Полученные данные принимаются с разной задержкой для каждой полосы из-за разного времени распространения, данные записываются в буфер выравнивания. В определенный момент времени (граница LMFC) все дорожки синхронно считываются из буфера.

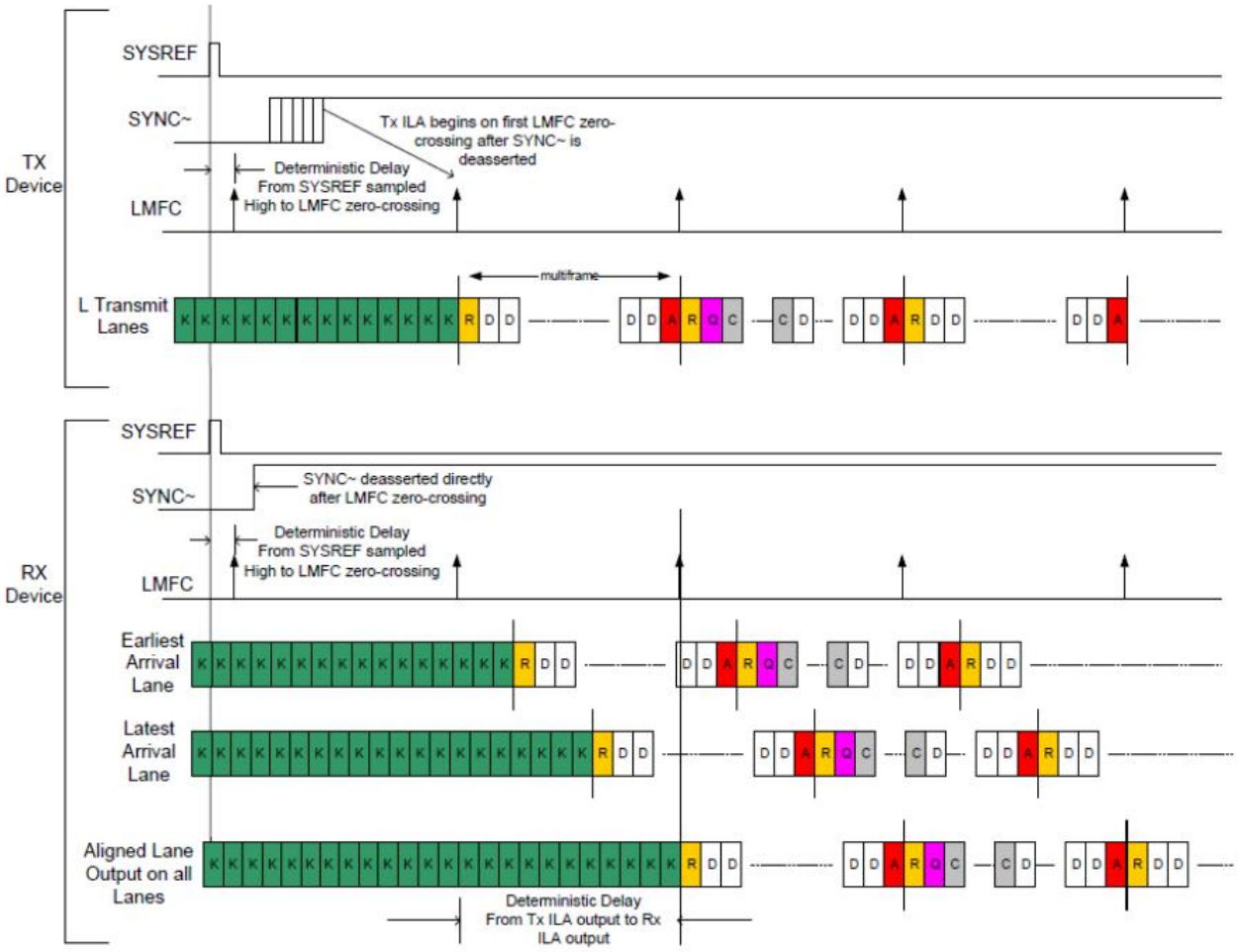


Рисунок 25.8. Синхронизации модуля приемника и передатчика (для операции Subclass 1)

На Рисунок 25.9 – представлена синхронизация модуля приемника и передатчика (Subclass 1) включающая RBD циклы кадров (RBD_LENGTH_OCTETS). RBD играет важную роль, чтение из буфера начнется после подсчета значения этого регистра в терминах октетов. Если, например, RBD_LENGTH_OCTETS равен 8, тогда чтение из буфера начнется через 2 такта clk_character после границы LMFC (поскольку канальный уровень обрабатывает 4 октета за такт). Если этот регистр установлен в 0, то чтение начнется в следующем такте clk_character после LMFC границы.

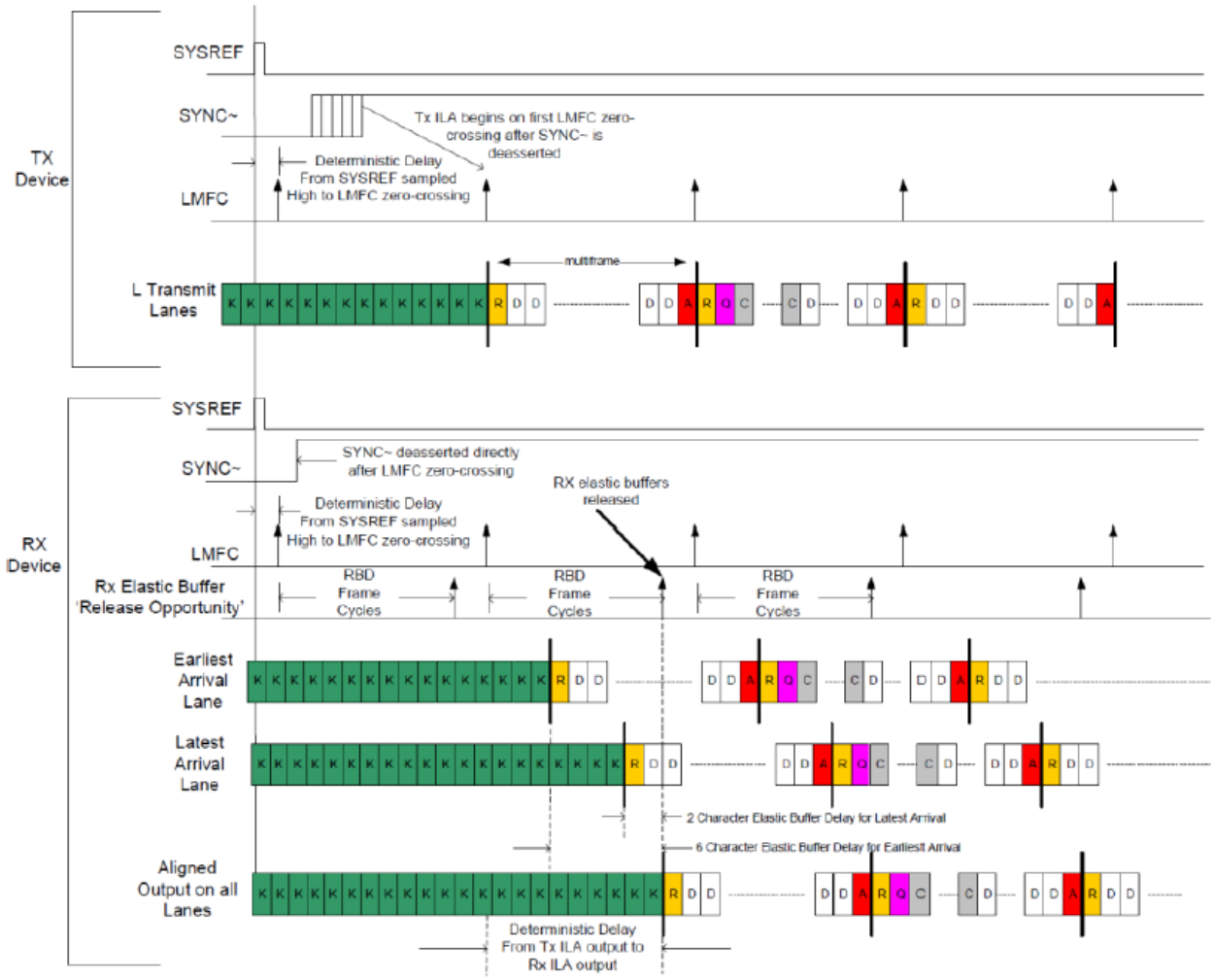


Рисунок 25.9. Синхронизация модуля приемника и передатчика (Subclass 1) включающая RBD циклы кадров

25.3.9.3 Subclass 2

Работа устройств подкласса 2 аналогична подклассу 1. Подкласс 2 поддерживает детерминированную задержку через интерфейс sync_b. Когда приемник выровнял входящие данные от передатчика, sync_b снимается. Передатчик обнаруживает изменение в сигнале и ждет следующего LMFC перед передачей последовательности ILA (в зависимости от регистра RX_FEATURE_DISABLE). Приемник буферизует данные, как описано на Рисунке 5 и Рисунка 6, прежде чем он доставит данные на интерфейс семплов.

25.3.10 Установка RBD_LENGTH_OCTETS

Регистр RBD_LENGTH_OCTETS используется в буфере выравнивания, расположенном в приемнике, и представляет задержку между границей LMFC и моментом, когда данные выводятся из выравнивающего буфера.

Диапазон значений регистра начинается от 0 до максимальной длины мультикадра, которая равна F * K и делится на 4 шага, так как канальный уровень Rx обрабатывает 4 слова каждый такт clk_character. Следовательно, значение регистра должен быть кратно 4.

Поступающие данные из serdes передаются в кодировщик и блок FS_SM. Блок FS_SM начинает отправку данных в выравнивающий буфер, когда ILAS обнаруживается после 4 символов K28.5.

В выравнивающем буфере расположен счетчик (считающий от 0 до F * K). Когда счетчик достигает значения, записанного в регистре RBD_LENGTH_OCTETS, контроллер проверяет наличие в буфер ILAS. Если ILAS не поступил, выравнивающий буфер будет ждать до следующей границы LMFC, когда счетчик снова достигнет значения RBD_LENGTH_OCTETS.

Учитывая это, значение RBD_LENGTH_OCTETS определяется относительной задержкой между прибытием ILAS в выравнивающий буфер и последней границей LMFC до этого.

RBD_LENGTH_OCTETS подсчитывается по следующей формуле:

> RBD_LENGTH_OCTETS = ([Внутренняя задержка TX] * 4 + [Внутренняя задержка RX] * 4 + [задержка передатчика (в циклах clk_character)] * 4 + [Максимальная задержка между полосами (в циклах clk_character)] * 4 + [физическая задержка (в циклах clk_character)] * 4 + [(время генерации LMFC в TX) - (время генерации LMFC в RX)] * 4) / (F * K)).

На рисунке 7. пример выравнивающего буфера и RBD_LENGTH_OCTETS. В примере работы выравнивающего буфера можем видеть что происходит, когда RBD_LENGTH_OCTETS равен 8. Регистр «current_char_cnt» считает от 0 до LMFC (до 32 в примере). Когда «current_char_cnt» достигает значения RBD_LENGTH_OCTETS, если ILAS уже прибыл, формируется «Deskew_ram_rd_en»подтверждающего возможность начать чтение буфера, куда уже поступили входные данные. Спустя два такта выровненные данные выводятся из блока «jesd_lane_deskew».

Чтобы рассчитать детерминированную задержку, необходимо учитывать задержку в блоках, где обрабатываются данные после выравнивающего буфера. Эта задержка показана в Табл. Задержка в RX (LINK LAYER).

25.3.11 Последовательность программирования и запуска

При включении питания сигнал «reset_core» должен в активном состоянии до тех пор, пока синхросигналы не станут стабильными. Если тактовая частота процессора уже стабильна, то reset_csr может быть снят, а регистры должны быть настроены до сброса reset_core. В текущей реализации тактовая частота процессора включена всегда, соответственно reset_csr можно снимать сразу.

Перед включением блока (ENABLEMODULE) нужно настроить конфигурационные регистры модуля приемника (CS, F, HD, K, L, M, N, NTOTAL, S и SCR).

25.3.11.1 Включение частот

Для работы контроллеров необходимо включить и корректно настроить (в соответствии с конфигурацией) следующие частоты:

- tx_clk_sample* – частота передачи семплов (источник UCG_JESD*_TX)
- tx_clk_character* – частота обработки данных (источник UCG_JESD*_TX)
- tx_clk_serdes* – частота передачи данных на PHY (источник pcie*_phy)
- rx_clk_sample – частота передачи семплов (источник UCG_JESD*_RX)
- rx_clk_character* – частота обработки данных (источник UCG_JESD*_RX)
- rx_clk_serdes[3:0] – частота приема данных с PHY (источник pcie*_phy)

* - 0 – для контроллера 0; 1 – для контроллера 1.

Для частот «sample» и «character» имеется возможность выбора референсного источника синхросигнала и его коэффициента деления. Адреса и описание регистров управления этими частотами представлены в Главе 6 «Подсистема SDR».

Частоты «serdes» формируются в физическом уровне блока PCIe. Принципы формирования этих частот описано в документе на физический уровень PCIe.

25.3.11.2 Управление сигналами сброса

Управление сигналами сброса для обоих контроллеров Jesd204b осуществляется через регистры «sdr_jesd0_rst_ctl» и «sdr_jesd1_rst_ctl». Адреса и описание регистров представлено в Главе 6 «Подсистема SDR».

25.3.11.3 Коммутация сигнала Sysref

Ввод в микросхему сигналов Sysref осуществляется только через две контактные площадки «JESD0_SYSREF» и «JESD1_SYSREF», при этом реализованная схема мультиплексии

позволяет подключить их к любому из четырех каналов синхронизации (Jesd0_tx_sysref, Jesd0_rx_sysref, Jesd1_tx_sysref, Jesd1_rx_sysref).

Управление коммутацией сигналами Sysref осуществляется через регистры «sdr_jesd0_ctl» и «sdr_jesd1_ctl». Адреса и описание регистров представлено в Главе 6 «Подсистема SDR».

25.4Схема подключения

25.4.1Jesd204b

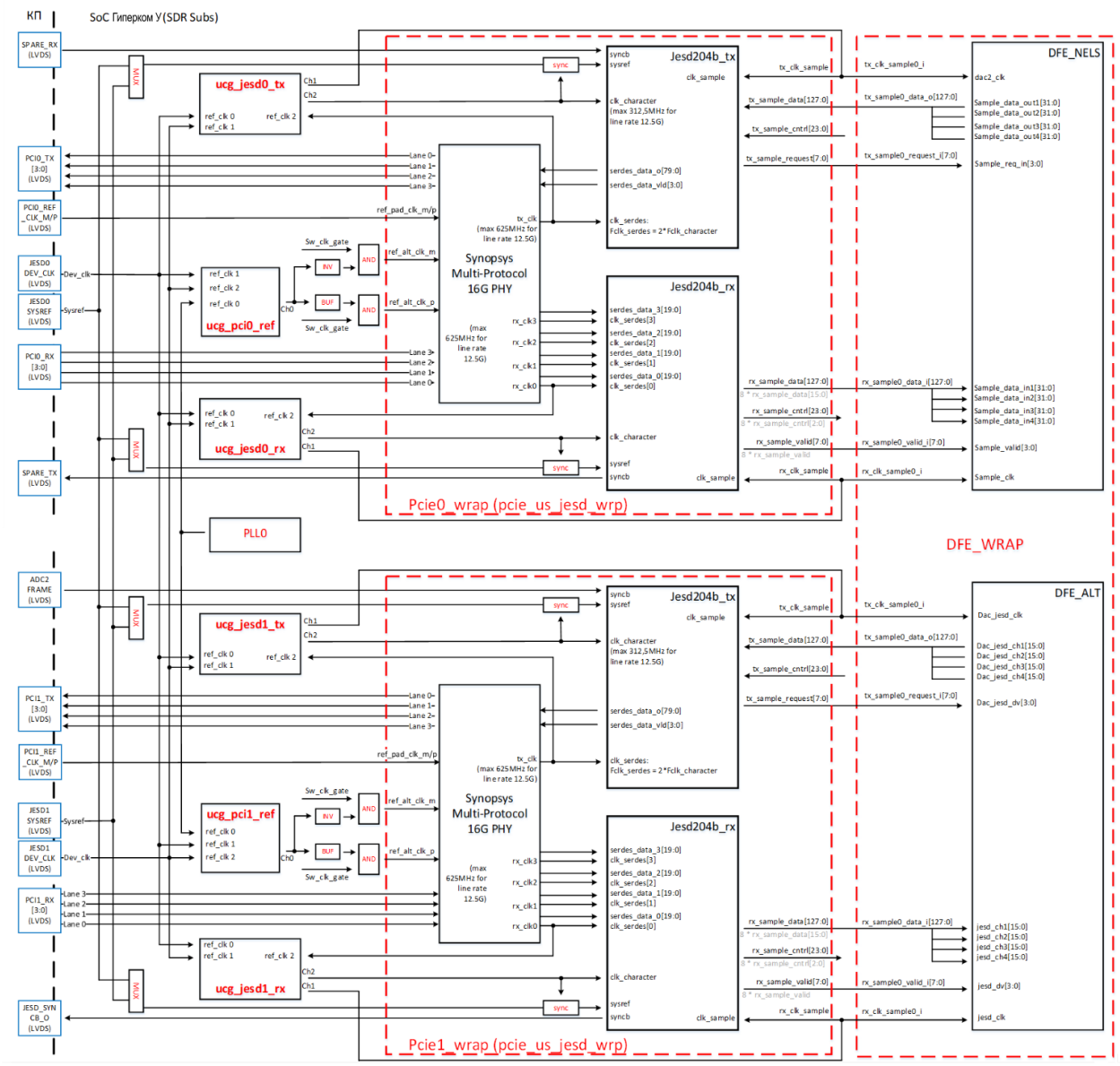


Рисунок 25.10. Схема подключения

26. КОНТРОЛЛЕР USB (USB)

26.1 Общая информация

Данный USB контроллер сконфигурирован как DRD (dual-role-device), то есть он может успешно выступать как в роли ведущего устройства (Host), так и в роли ведомого устройтсва (Device). Данное программируемое свойство предусмотренно для полной универсальности работы устройства в составе микросхемы.

Используемый контроллер USB полностью соответствует стандарту USB-3.0. Это означает что данный контроллер поддеживает следующий набор режимов скоростей для передачи данных: SuperSpeed (5 Gbps), High-speed (480Mbps) и Full-speed (12Mbps) в режиме работы “Device”, а также дополнительно и Low-speed (1.5Mbps) для режима работы “Host”.

Используемый контроллер USB-3.0 полностью обратно-совместим с контроллерами стандарта USB-2.0 для скоростных режимов High-speed, Full-speed и Low-speed. Функционал OTG (on-the-go) однако при этом для данного контроллера не предусмотрен.

Микросхема также снабжена встроенным совместимым вместе с данным USB контроллером USB-PHY обеспечивающим все необходимые высокоскоростные дифференциальные выводы соответсвующие стандарту USB3.0: двунаправленные DP и DM для поддержки скоростей предусмотренных стандартом 2.0, а также однонаравленные TX+ TX-, RX+ RX- для поддежки скоростного режима предусмотренного стандартом 3.0.

26.2 Блок регистров возможностей контроллера хоста USB3 eXtensible

Карта регистров DWC_usb3_block_eXtensible_Host_Cntrl_Cap_Regs приведена в Таблица 26.1.

Таблица 26.1. Карта регистров DWC_usb3_block_eXtensible_Host_Cntrl_Cap_Regs

№	Регистр	Описание	Смещение
1	CAPLENGTH	Длина регистров возможностей (Capability Registers Length)	0x0000
2	HCSPARAMS1	Регистр структурных параметров 1 (Structural Parameters 1 Register)	0x0004
3	HCSPARAMS2	Регистр структурных параметров 2 (Structural Parameters 2 Register)	0x0008
4	HCSPARAMS3	Регистр структурных параметров 3 (Structural Parameters 3 Register)	0x000C
5	HCCPARAMS1	Регистр параметров возможностей 1 (Capability Parameters 1 Register)	0x0010
6	DBOFF	Регистр смещения сигнала (Doorbell Offset Register)	0x0014
7	RTSOFF	Регистр смещения пробелов регистра выполнения (Runtime Register Space Offset Register)	0x0018
8	HCCPARAMS2	Параметры возможностей контроллера хоста 2 (Host Controller Capability Parameters 2)	0x001C

26.2.1 Длина регистров возможностей (CAPLENGTH)

Операционные регистры контроллера хоста = начальный адрес (Base address) + CAPLENGTH,

где CAPLENGTH – это`DWC_USB3_HOST_CAP_REG_LEN, чья длина по умолчанию – 20h.

Поля регистра CAPLENGTH описаны в Таблица 26.2.

Таблица 26.2. Длина регистров возможностей

Диапазон	Название	Описание	Исходное значение	Тип доступа
[7:0]	CAPLENGTH	Длина регистров возможностей (Capability Registers Length). Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x20	R
[15:8]	reserved_15_8	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0xff. Ограничение записи: writeAsRead.	0x0	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
[31:16]	HCIVERSION	Номер версии интерфейса RHC (RHC Interface Version Number). Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x110	R

26.2.2 Регистр структурных параметров 1 (HCSPARAMS1)

Описание регистра представлено в спецификации xHCI.

Поля регистра HCSPARAMS1 описаны в Таблица 26.3.

Таблица 26.3. Регистр структурных параметров 1

Диапазон	Название	Описание	Исходное значение	Тип доступа
[7:0]	MAXSLOTS	Количество слотов устройства (Number of device slots (MaxSlots)). Определяется конфигурируемым параметром `DWC_USB3_NUM_DEVICE_SUPT. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x40	R
[18:8]	MAXINTRS	Количество прерывателей (Number of Interrupters). Определяется конфигурируемым параметром `DWC_USB3_HOST_NUM_INTERRUPTER_SUPT Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x6	R
[23:19]	reserved_23_19	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0x1f. Ограничение записи: writeAsRead.	0x0	R
[31:24]	MAXPORTS	Количество портов (Number of Ports). Количество реализованных портов определяется параметром (`DWC_USB3_HOST_NUM_U2_ROOT_PORTS + `DWC_USB3_HOST_NUM_U3_ROOT_PORTS). Количество подключенных портов контролируется сигналами ввода системы host_num_u2_port[3:0]+host_num_u3_port[3:0]. Замечание: в режиме работы только с USB 2.0, сигнал host_num_u3_port является нулевым. Контролепригодность: нетестируемо. Ограничение записи: writeAsRead.	0x2	R

26.2.3 Регистр структурных параметров 2 (HCSPARAMS2)

Описание регистра представлено в спецификации xHCI.

Поля регистра HCSPARAMS2 описаны в Таблица 26.4.

Таблица 26.4. Регистр структурных параметров 2

Диапазон	Название	Описание	Исходное значение	Тип доступа
[3:0]	IST	Граничное значение изохронного планирования (Isochronous Scheduling Threshold). Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x1	R
[7:4]	ERSTMAX	Максимальное значение таблицы сегмента кольца (кольцевого буфера) событий (Event Ring Segment Table Max). Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0xf	R
[20:8]	reserved_20_8	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0x1fff. Ограничение записи: writeAsRead.	0x0	R
[25:21]	MAXSCRATCHPADBUFS_HI	Старшие биты максимального значения буферов блокнотной памяти (Max Scratchpad Bufs HI). Система обновляет это поле автоматически. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R
[26]	SPR	Восстановление блокнотной памяти (Scratchpad Restore). Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x1	R
[31:27]	MAXSCRATCHPADBUFS	Младшие биты максимального значения буферов блокнотной памяти (Max Scratchpad Bufs Lo). Это значение вычисляется на основе выбранных значений параметра конфигурации. Возможные значения: 1-4. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x1	R

26.2.4 Регистр структурных параметров 3 (HCSPARAMS3)

Описание регистра представлено в спецификации xHCI.

Поля регистра HCSPARAMS3 описаны в Таблица 26.5.

Таблица 26.5. Регистр структурных параметров 3

Диапазон	Название	Описание	Исходное значение	Тип доступа
[7:0]	U1_DEVICE_EXIT_LAT	Запаздывание выхода устройства из U1 (U1 Device Exit Latency). Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0xa	R
[15:8]	reserved_15_8	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0xff. Ограничение записи: writeAsRead.	0x0	R
[31:16]	U2_DEVICE_EXIT_LAT	Запаздывание выхода устройства из U2 (U2 Device Exit Latency). Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x7ff	R

26.2.5 Регистр параметров возможностей 1 (HCCPARAMS1)

Описание регистра представлено в спецификации xHCI.

Поля регистра HCCPARAMS1 описаны в Таблица 26.6.

Таблица 26.6. Регистр параметров возможностей 1

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	AC64	Возможность 64-битной адресации (64-bit Addressing Capability). Описание этого поля стандартного регистра USB представлено в спецификации интерфейса контроллера хоста eXtensible для USB 3.0. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x1	R
[1]	BNC	Возможность согласования разрешения полосы частот (RBW Negotiation Capability). Описание этого поля стандартного регистра USB представлено в спецификации интерфейса контроллера хоста eXtensible для USB 3.0. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R
[2]	CSZ	Размер контекста (Context Size). Описание этого поля стандартного регистра USB представлено в спецификации интерфейса контроллера хоста eXtensible для USB 3.0. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x1	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
[3]	PPC	Управление питанием порта (Port Power Control). Описание этого поля стандартного регистра USB представлено в спецификации интерфейса контроллера хоста eXtensible для USB 3.0. Контролепригодность: нетестируемо. Изменчивость: true. Ограничение записи: writeAsRead.	0x1	R
[4]	PIND	Индикаторы порта (Port Indicators). Описание этого поля стандартного регистра USB представлено в спецификации интерфейса контроллера хоста eXtensible для USB 3.0. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R
[5]	LHRC	Возможность легкого сброса контроллера хоста (Light HC Reset Capability). Описание этого поля стандартного регистра USB представлено в спецификации интерфейса контроллера хоста eXtensible для USB 3.0. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x1	R
[6]	LTC	Возможность допуска запаздывания обмена сообщениями (Latency Tolerance Messaging Capability). Описание этого поля стандартного регистра USB представлено в спецификации интерфейса контроллера хоста eXtensible для USB 3.0. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x1	R
[7]	NSS	Вторичная поддержка SID отсутствует (No Secondary SID Support). Описание этого поля стандартного регистра USB представлено в спецификации интерфейса контроллера хоста eXtensible для USB 3.0. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R
[8]	PAE	Парсинг всех данных о событиях (Parse All Event Data). Описание этого поля стандартного регистра USB представлено в спецификации интерфейса контроллера хоста eXtensible для USB 3.0. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
[9]	SPC	Возможность короткого пакета (Short Packet Capability). Описание этого поля стандартного регистра USB представлено в спецификации интерфейса контроллера хоста eXtensible для USB 3.0. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x1	R
[10]	SEC	Возможность остановки EDLTA (Stopped EDLTA Capability). Описание этого поля стандартного регистра USB представлено в спецификации интерфейса контроллера хоста eXtensible для USB 3.0. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x1	R
[11]	CFC	Возможность идентификатора смежного фрейма (Contiguous Frame ID Capability). Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x1	R
[15:12]	MAXPSASIZE	Максимальный размер массива основного потока (Maximum Primary Stream Array Size). Описание этого поля стандартного регистра USB представлено в спецификации интерфейса контроллера хоста eXtensible для USB 3.0. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0xf	R
[31:16]	XECP	Указатель расширенных возможностей xHCI (xHCI Extended Capabilities Pointer). Система автоматически обновляет его на основе конфигурации. Подробнее по DWC_USB3_HC_XECP следует обращаться к <workspace>/src/DWC_usb3_params.v. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x248	R

26.2.6 Регистр смещения сигнала (DBOFF)

Описание регистра представлено в спецификации xHCI.

Поля регистра DBOFF описаны в Таблица 26.7.

Таблица 26.7. Регистр смещения сигнала

Диапазон	Название	Описание	Исходное значение	Тип доступа
[1:0]	reserved_1_0	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0x3. Ограничение записи: writeAsRead.	0x0	R
[31:2]	DOORBELL_ARRAY_OFFSET	Смещение массива сигнала – только чтение (Doorbell Array Offset – RO). Система автоматически обновляет это поле на основе конфигурации. Описание этого поля стандартного регистра USB представлено в спецификации интерфейса контроллера хоста eXtensible для USB 3.0. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x148	R

26.2.7 Регистр пространственного смещения регистра выполнения (RTSOFF)

Поля регистра RTSOFF описаны в Таблица 26.8.

Таблица 26.8. Регистр пространственного смещения регистра выполнения

Диапазон	Название	Описание	Исходное значение	Тип доступа
[4:0]	reserved_4_0	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0x1f. Ограничение записи: writeAsRead.	0x0	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
[31:5]	RUNTIME_REG_SPACE_OFFSET	Пространственное смещение регистра выполнения (Runtime Register Space Offset). Система автоматически обновляет это поле на основе конфигурации. Описание этого поля стандартного регистра USB представлено в спецификации интерфейса контроллера хоста eXtensible для USB 3.0. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x22	R

26.2.8 Параметры возможностей контроллера хоста 2 (HCCPARAMS2)

Описание регистра представлено в спецификации xHCI.

Поля регистра HCCPARAMS2 описаны в Таблица 26.9.

Таблица 26.9. Параметры возможностей контроллера хоста 2

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	U3C	Возможность входа в U3 (U3 Entry Capability). Описание этого поля стандартного регистра USB представлено в спецификации интерфейса контроллера хоста eXtensible для USB 3.0. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x1	R
[1]	CMC	Возможность слишком большого максимального запаздывания выхода команды конфигурирования конечной точки (Configure Endpoint Command Max Exit Latency Too Large Capability). Описание этого поля стандартного регистра USB представлено в спецификации интерфейса контроллера хоста eXtensible для USB 3.0. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x1	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
[2]	FSC	Возможность принудительного сохранения контекста (Force Save Context Capability). Описание этого поля стандартного регистра USB представлено в спецификации интерфейса контроллера хоста eXtensible для USB 3.0. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x1	R
[3]	CTC	Возможность переключения совместимости (Compliance Transition Capability). Описание этого поля стандартного регистра USB представлено в спецификации интерфейса контроллера хоста eXtensible для USB 3.0. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x1	R
[4]	LEC	Возможность большой полезной нагрузки интервала обслуживания конечной точки (endpoint service interval time, ESIT) (Large ESIT Payload Capability). Описание этого поля стандартного регистра USB представлено в спецификации интерфейса контроллера хоста eXtensible для USB 3.0. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R
[5]	CIC	Возможность информации о конфигурации (Configuration Information Capability (CIC)). Описание этого поля стандартного регистра USB представлено в спецификации интерфейса контроллера хоста eXtensible для USB 3.0. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x1	R
[31:6]	reserved_31_6	Зарезервировано. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R

26.3 Блок глобальных регистров USB3

Карта регистров DWC_usb3_block_gbl приведена в Таблица 26.10.

Таблица 26.10. Карта регистров DWC_usb3_block_gbl

№	Регистр	Описание	Смещение
1	GSBUSCFG0	Глобальный регистр конфигурирования шины SoC 0 (Global SoC Bus Configuration Register 0)	0xc100
2	GSBUSCFG1	Глобальный регистр конфигурирования шины SoC 1 (Global SoC Bus Configuration Register 1)	0xC104
3	GTXTHRCFG	Глобальный регистр настроек пороговых значений Tx (Global Tx Threshold Control Register)	0xc108
4	GRXTHRCFG	Глобальный регистр настроек пороговых значений Rx (Global Rx Threshold Control Register)	0xc10c
5	GCTL	Глобальный регистр системных настроек (Global Core Control Register)	0xc110
6	GPMSTS	Глобальный регистр статуса управления питанием (Global Power Management Status Register)	0xc114
7	GSTS	Глобальный регистр статуса (Global Status Register)	0xc118
8	GUCTL1	Глобальный регистр настроек пользователя 1 (Global User Control Register 1)	0xc11c
9	GSNPSID	Глобальный регистр идентификатора Synopsys (Global Synopsys ID Register)	0xc120
10	GGPIO	Глобальный регистр ввода/вывода общего назначения (Global General Purpose Input/Output Register)	0xc124
11	GUID	Глобальный регистр идентификатора пользователя (Global User ID Register)	0xc128
12	GUCTL	Глобальный регистр пользовательских настроек (Global User Control Register)	0xc12c
13	GBUSERRADDRLO	Младший глобальный регистр ошибок адреса шины SoC (Gobal SoC Bus Error Address Register – Low)	0xc130
14	GBUSERRADDRHI	Старший глобальный регистр ошибок адреса шины SoC (Gobal SoC Bus Error Address Register – High)	0xc134
15	GPRTBIMAPLO	Младший глобальный регистр установки соответствий сверхскоростного порта к экземпляру шины (Global SS Port to Bus Instance Mapping Register – Low)	0xc138

№	Регистр	Описание	Смещение
16	GPRTBIMAPHI	Старший глобальный регистр установки соответствий сверхскоростного порта к экземпляру шины (Global SS Port to Bus Instance Mapping Register – High)	0xc13c
17	GHWPARAMS0	Глобальный регистр аппаратных параметров 0 (Global Hardware Parameters Register 0)	0xc140
18	GHWPARAMS1	Глобальный регистр аппаратных параметров 1 (Global Hardware Parameters Register 1)	0xc144
19	GHWPARAMS2	Глобальный регистр аппаратных параметров 2 (Global Hardware Parameters Register 2)	0xc148
20	GHWPARAMS3	Глобальный регистр аппаратных параметров 3 (Global Hardware Parameters Register 3)	0xc14c
21	GHWPARAMS4	Глобальный регистр аппаратных параметров 4 (Global Hardware Parameters Register 4)	0xc150
22	GHWPARAMS5	Глобальный регистр аппаратных параметров 5 (Global Hardware Parameters Register 5)	0xc154
23	GHWPARAMS6	Глобальный регистр аппаратных параметров 6 (Global Hardware Parameters Register 6)	0xc158
24	GHWPARAMS7	Глобальный регистр аппаратных параметров 7 (Global Hardware Parameters Register 7)	0xc15c
25	GDBGFIFOSPACE	Глобальный регистр доступности пространства буфера FIFO/очереди отладки (Global Debug Queue/FIFO Space Available Register)	0xc160
26	GDBGLTSSM	Глобальный регистр отладки LTSSM (Global Debug LTSSM Register)	0xc164
27	GDBGLNMCC	Глобальный регистр отладки LNMCC (Global Debug LNMCC Register)	0xc168
28	GDBGBMU	Глобальный регистр отладки BMU (Buffer Mangement Unit – модуль управления буферами) (Global Debug BMU Register)	0xc16c
29	GDBGLSPMUX_DEV	Глобальный регистр отладки LSP MUX для устройства (Global Debug LSP MUX Register – Device)	0xc170
30	GDBGLSPMUX_HST	Глобальный регистр отладки LSP MUX для хоста (Global Debug LSP MUX Register – Host)	0xc170
31	GDBGLSP	Глбальный регистр отладки LSP (Global Debug LSP Register)	0xC174
32	GDBGEPINFO0	Глобальный регистр информации по отладке конечной точки 0 (Global Debug Endpoint Information Register 0	0xC178

№	Регистр	Описание	Смещение
33	GDBGEPINFO1	Глобальный регистр информации по отладке конечной точки 1 (Global Debug Endpoint Information Register 1)	0xc17c
34	GPRTBIMAP_HSLO	Младший глобальный регистр установки соответствий высокоскоростного порта к экземпляру шины (Global High–Speed Port to Bus Instance Mapping Register – Low)	0xc180
35	GPRTBIMAP_HSHI	Старший глобальный регистр установки соответствий высокоскоростного порта к экземпляру шины (Global High–Speed Port to Bus Instance Mapping Register – High)	0xc184
36	GPRTBIMAP_FSLO	Младший глобальный регистр установки соответствий полноскоростного порта к экземпляру шины (Global Full–Speed Port to Bus Instance Mapping Register – Low)	0xc188
37	GPRTBIMAP_FSHI	Старший глобальный регистр установки соответствий полноскоростного порта к экземпляру шины (Global Full–Speed Port to Bus Instance Mapping Register – High)	0xc18c
38	GUCTL2	Глобальный регистр настроек пользователя 2 (Global User Control Register 2)	0xc19c
39	GUSB2PHYCFG(#n) (for n = 0; n <= DWC_USB3_NUM_U2_ROOT_PORTS-1)	Глобальный регистр конфигурирования PHY USB2 (Global USB2 PHY Configuration Register)	0xc200
40	GUSB2I2CCTL(#n)(for n == 0)	Зарезервированный регистр (Reserved Register)	0xc240
41	GUSB2PHYACC_ULPI(#n)(for n == 0)	Глобальный регистр управления USB 2.0 ULPI PHY от поставщика (Global USB 2.0 ULPI PHY Vendor Control Register)	0xc280
42	GUSB2PHYACC_UTMI(#n)(for n == 0)	Глобальный регистр управления USB 2.0 UTMI PHY от поставщика (Global USB 2.0 UTMI PHY Vendor Control Register)	0xc280
43	GUSB3PIPECTL(#n) (for n = 0; n <= DWC_USB3_NUM_U3_ROOT_PORTS-1)	Глобальный регистр настроек PIPE USB 3.0 (Global USB 3.0 PIPE Control Register)	0xc2c0
44	GTXFIFOSIZ(#n)(for n = 0; n <= 31)	Глобальный регистр размера буфера FIFO на передачу (Global Transmit FIFO Size Register)	0xc300
45	GRXFIFOSIZ(#n)(for n = 0; n <= 31)	Глобальный регистр размера буфера FIFO на получение (Global Receive FIFO Size Register)	0xc380

№	Регистр	Описание	Смещение
46	GEVNTADRLO(#n)(for n == 0)	Младший глобальный регистр адреса буфера событий (Global Event Buffer Address (Low) Register)	0xc400
47	GEVNTADRHI(#n)(for n == 0)	Старший глобальный регистр адреса буфера событий (Global Event Buffer Address (High) Register)	0xc404
48	GEVNTSIZ(#n)(for n == 0)	Глобальный регистр размера буфера событий (Global Event Buffer Size Register)	0xc408
49	GEVNTCOUNT(#n)(for n == 0)	Глобальный регистр счетчика буфера событий (Global Event Buffer Count Register)	0xc40c
50	GHWPARAMS8	Глобальный регистр аппаратных параметров 8 (Global Hardware Parameters Register 8)	0xc600
51	GTXFIFOPRIDEV	Глобальный регистр приоритета DMA для буферов FIFO TX устройства (Global Device TX FIFO DMA Priority Register)	0xc610
52	GTXFIFOPRIHST	Глобальный регистр приоритета DMA для буферов FIFO TX хоста (Global Host TX FIFO DMA Priority Register)	0xc618
53	GRXFIFOPRIHST	Глобальный регистр приоритета DMA для буферов FIFO RX хоста (Global Host RX FIFO DMA Priority Register)	0xc61c
54	GFIFOPRIDBC	Глобальный регистр приоритета DMA для возможностей отладки (Global Host Debug Capability DMA Priority Register).	0xc620
55	GDMAHLRATIO	Глобальный регистр соотношения высокого и низкого приоритетов DMA для буферов FIFO в режиме хоста (Global Host FIFO DMA High-Low Priority Ratio Register)	0xc624
56	GFLADJ	Глобальный регистр настройки длины фреймов (Global Frame Length Adjustment Register)	0xc630

26.3.1 Глобальный регистр конфигурирования шины SoC 0 (GSBUSCFG0)

Этот регистр конфигурирует опции прямого доступа системной шины к памяти (DMA) для главной шины, которая может конфигурироваться как AHB, AXI, или Native (встроенная). Опции включают длину импульса и тип кэша (буферизуемый/доставляемый, кэшируемый/следящий (bufferable/posted, cacheable/snoop) и т.д.). Приложение может запрограммировать этот регистр при запуске, или при смене режима работы после того, как механизм DMA приостановлен.

Значение регистра xHCI при запуске.

Если используется стандартный драйвер хоста xHCI, следует выставить значение этого регистра при запуске отдельно, во время конфигурирования в coreConsultant (параметр DWC_USB3_GSBUSCFG0_INIT), поскольку стандартный драйвер xHCI не имеет доступа к этому регистру.

Подробнее об этом регистре написано в главе «Детали архитектуры» в справочнике по сверхскоростному контроллеру USB 3.0 DWC.

Поля регистра GSBUSCFG0 описаны в Таблица 26.11.

Таблица 26.11. Глобальный регистр конфигурирования шины SoC 0

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	INCRBRSTENA	Подключение типа импульса INCR неопределенной длины (Undefined Length INCR Burst Type Enable (INCRBrstEna) Ввод шины BUS-GM; Этот бит определяет набор длин импульсов, используемый главным интерфейсом. Он работает в сочетании с GSBUSCFG0[7:1] и подключает (INCR256/128/64/32/16/8/4). 0: импульсный режим INCRX. В HBURST (для конфигураций AHB) и ARLEN/AWLEN (для конфигураций AXI) импульсы INCR не используются, кроме случаев невыровненных передач импульса. В случае передач, выровненных по адресу, в них используются только следующие длины импульсов: - 1; - 4 (если GSBUSCFG0.INCR4BrstEna = 1); - 8 (если GSBUSCFG0.INCR8BrstEna = 1); - 16 (если GSBUSCFG0.INCR16BrstEna = 1); - 32 (если GSBUSCFG0.INCR32BrstEna = 1); - 64 (если GSBUSCFG0.INCR64BrstEna = 1); - 128 (если GSBUSCFG0.INCR128BrstEna = 1); - 256 (если GSBUSCFG0.INCR256BrstEna = 1). Замечания: - в случае передач, не выровненных по адресу, импульс INCR может быть сгенерирован в начале и конце передач, чтобы выровнять границы адресов, даже если INCR отключен. - в режиме AHB, если подключен режим импульсов INCRX, но не подключен ни один из поддерживаемых INCRx битов импульсов, контроллер будет производить импульсы INCR неопределенной длины. 1: режим импульсов INCR (неопределенной длины). Конфигурации AHB: в HBURST используется импульс SINGLE или INCR любой длины с обработкой превышения границ в 1 Кбайт.	0x1	R/W

		Конфигурации AXI: в ARLEN/AWLEN используется любая длина, меньшая или равная наибольшей из подключенных длин импульса INCR32/64/128/256. Для приложений кэша, выровненных по длине линии, в этот бит, как правило, установлен 0, чтобы гарантировать, что главный интерфейс использует только длины импульса степеней двойки (подключенные через GSBUSCFG0[7:0]).		
[1]	INCR4BRSTENA	Подключение типа импульса INCR4. Ввод шины BUS-GM; В конфигурациях AXI, когда этот бит подключен, контроллеру позволено производить импульсы только длиной биения (биение, beat length - длина отдельной передачи данных внутри импульса) 1, 2, 3 и 4. Настоятельно рекомендуется подключать этот бит, чтобы предотвратить разбиение чтений и записей дескриптора на раздельные передачи.	0x0	R/W
[2]	INCR8BRSTENA	Подключение типа импульса INCR8. Ввод шины BUS-GM; В конфигурации AHB/AXI, если программное обеспечение устанавливает «1» в этот бит, главный интерфейс AHB/AXI использует INCR для исполнения импульса на 8 биений (на 8 передач данных внутри одного импульса - 8-beat).	0x0	R/W
[3]	INCR16BRSTENA	Подключение типа импульса INCR16. Ввод шины BUS-GM; В конфигурации AHB/AXI, если программное обеспечение устанавливает «1» в этот бит, главный интерфейс AHB/AXI использует INCR для исполнения импульса на 16 биений (на 16 передач данных внутри одного импульса - 16-beat).	0x0	R/W
[4]	INCR32BRSTENA	Подключение типа импульса INCR32. Ввод шины BUS-GM; В конфигурации AHB/AXI, если программное обеспечение устанавливает «1» в этот бит, главный интерфейс AHB/AXI использует INCR для исполнения импульса на 32 биения (на 32 передачи данных внутри одного импульса - 32-beat).	0x0	R/W
[5]	INCR64BRSTENA	Подключение типа импульса INCR64. Ввод шины BUS-GM; В конфигурации AHB/AXI, если программное обеспечение устанавливает «1» в этот бит, главный интерфейс AHB/AXI использует INCR для исполнения импульса на 64 биения (на 64 передачи данных внутри одного импульса - 64-beat).	0x0	R/W
[6]	INCR128BRSTENA	Подключение типа импульса INCR128. Ввод шины BUS-GM; В конфигурации AHB/AXI, если программное обеспечение устанавливает «1» в этот бит, главный интерфейс AHB/AXI использует INCR для исполнения импульса на 128 биений (на 128 передач данных внутри одного импульса - 128-beat).	0x0	R/W
[7]	INCR256BRSTENA	Подключение типа импульса INCR128. Ввод шины BUS-GM;	0x0	R/W

		В конфигурации AHB/AXI, если программное обеспечение устанавливает «1» в этот бит, главный интерфейс AHB/AXI использует INCR для исполнения импульса на 256 биений (на 256 передач данных внутри одного импульса - 256-beat).		
[9:8]	reserved_9_8	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0x3. Ограничение записи: writeAsRead.	0x0	R
[10]	DESBIGEND	Доступ к дескриптору: от старшего бита к младшему (в обратном порядке). Этот бит управляет порядком битов для обращений к дескриптору. - прямой порядок (по умолчанию). - обратный порядок. В режиме обратного порядка битов, обращение DMA (как на чтение, так и на запись) к дескрипторам пользуется режимом обратного порядка с неизменными байтами (Byte Invariant Big-Endian) (см. раздел «Прямой и обратный порядок» в руководстве пользователя сверхскоростного контроллера USB 3.0 DWC). Данные в дескрипторах интерпретируются как «встроенные данные» в следующих случаях: - в режиме “Device”: указатель буфера блока запуска Setup TRB указывает на сам Setup TRB. - в режиме “Host”: в бит непосредственно получаемых данных (The Immediate Data (IDT) bit) в блоке передачи Transfer TRB установлена 1. В режиме “Device”, если система использует для дескриптора и для данных разные режимы порядка байтов, программное обеспечение не должно пользоваться «встроенными» данными. В режиме “Host”, если система использует для дескриптора и для данных разные режимы порядка байтов, контроллер обрабатывает «встроенные данные» как дескриптор (а не как данные) в смысле обработки режима порядка байтов. Если только система не ожидает именно этого, программное обеспечение должно обрабатывать «встроенные данные» соответствующим образом. Замечание: поскольку интерфейс AXI требует режим порядка битов с неизменными байтами, установка 1 в DescBigend и DatBigEnd провоцирует применение трансформации неизменности адреса (address invariant transform), что нежелательно. Подробнее см. Разделы 9.3 и 9.4 Спецификации AMBA AXI. Таким образом, для главного интерфейса AXI (DWC_USB3_MBUS_TYPE=1) в этот бит необходимо устанавливать 0.	0x0	R/W

[11]	DATBIGEND	Доступ к данным: от старшего бита к младшему (в обратном порядке). Этот бит управляет порядком битов для обращений к данным. - прямой порядок (по умолчанию); - обратный порядок. В режиме обратного порядка битов, обращение DMA (как на чтение, так и на запись) к пакету данных пользуется режимом обратного порядка с неизменными байтами (Byte Invariant Big-Endian) (см. Раздел «Прямой и обратный порядок» в руководстве пользователя сверхскоростного контроллера USB 3.0 DWC). Замечание: поскольку интерфейс AXI требует режима порядка битов с неизменными байтами, установка 1 в DescBigend и DatBigEnd провоцирует применение трансформации неизменности адреса (address invariant transform), что нежелательно. Подробнее см. Разделы 9.3 и 9.4 Спецификации AMBA AXI. Таким образом, для главного интерфейса AXI (DWC_USB3_MBUS_TYPE=1) в этот бит необходимо устанавливать 0.	0x0	R/W
[15:12]	reserved_15_12	Зарезервировано для дальнейшего использования. Контролепригодность: writeAsRead. Маска сброса: 0xf. Ограничение записи: writeAsRead.	0x0	R
[19:16]	DESWRREQINFO	AHB-prot/AXI-cache/OCP-ReqInfo для записей дескриптора (DesWrReqInfo). Ввод шины BUS-GM.	0x0	R/W
[23:20]	DATWRREQINFO	AHB-prot/AXI-cache/OCP-ReqInfo для записей данных (DatWrReqInfo). Ввод шины BUS-GM.	0x0	R/W
[27:24]	DESRDREQINFO	AHB-prot/AXI-cache/OCP-ReqInfo для чтений дескриптора (DesRdReqInfo). Ввод шины BUS-GM.	0x0	R/W
[31:28]	DATRDREQINFO	AHB-prot/AXI-cache/OCP-ReqInfo для чтений данных (DatRdReqInfo). Ввод шины BUS-GM.	0x0	R/W

26.3.2 Глобальный регистр конфигурирования шины SoC 1 (GSBUSCFG1)

Значение регистра xHCI при запуске: если используется стандартный драйвер хоста, следует задать значение регистра при запуске во время конфигурирования в coreConsultant (параметр DWC_USB3_GSBUSCFG1_INIT), потому что стандартный драйвер xHCI не обращается к этому регистру.

Подробнее об этом регистре написано в главе «Детали архитектуры» в справочнике по сверхскоростному контроллеру USB 3.0 DWC.

Поля регистра GSBUSCFG1 описаны в Таблица 26.12.

Таблица 26.12. Глобальный регистр конфигурирования шины SoC 1

Диапазон	Название	Описание	Исходное значение	Тип доступа
[7:0]	reserved_7_0	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0xff. Ограничение записи: writeAsRead.	0x0	R
[11:8]	PipeTransLimit	Ограничение на запросы об импульсах конвейерной передачи AXI (AXI Pipelined Transfers Burst Request Limit). Это поле управляет количеством добавочных запросов на конвейерную передачу, которые основное устройство (master) AXI отправляет подчиненному устройству (slave). Когда основное устройство достигает этого лимита, оно больше не отправляет запросы на шины ARADDR и AWADDR AXI до тех пор, пока не завершатся соответствующие фазы данных. Кодирование значений поля: 'h0: 1 запрос; 'h1: 2 запроса; 'h2: 3 запроса; 'h3: 4 запроса; 'hF: 16 запросов.	0x3	R/W
[12]	EN1KPAGE	Подключение ограничения в 1000 страниц (1k Page Boundary Enable). По умолчанию (когда этот бит отключен) AXI прекращает передачу, когда достигает ограничения в 4000 страниц. Когда этот бит подключен, основное устройство AXI (данные DMA) прекращает передачи по достижении ограничения в 1000 страниц.	0x0	R/W
[31:13]	reserved_31_13	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0x7ffff. Ограничение записи: writeAsRead.	0x0	R

26.3.3 Глобальный регистр настроек пороговых значений Tx (GTXTHRCFG)

Более подробно применение этого регистра описано в главе «Детали архитектуры».

Способы выбора значений для полей этого регистра описаны подробнее в Разделе «Размеры буферов данных FIFO TX/RX и настройки регистра управления пороговыми значениями TX/RX» в Руководстве пользователя контроллера сверхскоростной USB 3.0 DWC.

Замечания:

регистр GTXTHRCFG неприменим для целевой отладки (Debug Target).
регистр GTXTHRCFG неприменим строго в рамках режима USB 2.0.

Поля регистра GTXTHRCFG описаны в Таблица 26.13.

Таблица 26.13. Глобальный регистр настроек пороговых значений Tx

Диапазон	Название	Описание	Исходное значение	Тип доступа
[10:0]	reserved_10_0	Зарезервировано для дальнейшего использования. Контролепригодность: writeAsRead. Маска сброса: 0x7ff. Ограничение записи: writeAsRead.	0x0	R
[13:11]	reserved_13_11	Зарезервировано (Rsvd/Rs). Приложениям следует писать в это поле только нулевое значение. Прочтенные отсюда значения должны трактоваться как X (неизвестные). Контролепригодность: writeAsRead. Маска сброса: 0x7. Ограничение записи: writeAsRead.	0x0	R
[14]	reserved_14	Зарезервировано (Rsvd/Rs). Приложениям следует писать в это поле только нулевое значение. Прочтенные отсюда значения должны трактоваться как X (неизвестные). Контролепригодность: writeAsRead. Маска сброса: 0x1. Ограничение записи: writeAsRead.	0x0	R
[15]	reserved_15	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0x1. Ограничение записи: writeAsRead.	0x0	R
[23:16]	UsbMaxTxBurstSize	Максимальный размер импульса TX в USB (USB Maximum TX Burst Size). Когда поле UsbTxPktCntSel имеет значение «1», это поле задает максимальный размер импульса типа Bulk OUT, который система способна выдать. Когда системная шина медленнее, чем шина USB, буфер TX FIFO может оказаться недозаполнен во время длинного импульса. Пользователь может запрограммировать в это поле меньшее значение для того, чтобы ограничить размер импульса TX, который может выдать система. Режим “Host”: применимо только к конечным точкам типов SS Bulk, Interrupt OUT, и Isochronous. Режим “Device”: это значение не используется в режиме “Device”, но пользователям необходимо запрограммировать значение во время использования функции ограничения TX (threshold feature) для того, чтобы запрограммированная в UsbTxPktCnt величина была точно меньше, чем эта. Валидные значения: от 1 до 16. Контролепригодность: неограниченно.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
[27:24]	UsbTxPktCnt	Счётчик переданных пакетов USB (USB Transmit Packet Count). Это поле задает число пакетов, которое должно быть загружено в буфер TXFIFO прежде, чем система может начать передачу для соответствующей USB-транзакции (импульса). Это поле валидно только когда в поле подключения счетчика переданных пакетов USB установлено значение «1». Валидные значения: от 1 до 15. Замечание: - в режиме “Device”, если контроллер устройства не обладает соответствующим количеству пакетов числом блоков TRB или не может вызвать блоки TRB из-за сильного запаздывания или переключения между остальными конечными точками, тогда он не станет ожидать порогового числа пакетов (threshold number of packets). Пороговое число пакетов будет учтено только если в контроллере доступно соответствующее числу пакетов число блоков TRB до того, как он начнёт выборку данных. - это поле должно иметь значение большее или равное значению поля максимального размера импульса TX USB. Контролепригодность: неограниченно.	0x0	R/W
[28]	reserved_28	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0x1. Ограничение записи: writeAsRead.	0x0	R
[29]	UsbTxPktCntSel	Подключение счётчика переданных пакетов USB (USB Transmit Packet Count Enable). Это поле подключает/отключает пороговые значения многопакетной передачи USB (multi-packet thresholding): 0: пороговые значения многопакетной передачи USB отключены, система может начинать передачу на USB после того, как один (полный) пакет будет целиком загружен в соответствующий буфер TXFIFO. 1: пороговые значения многопакетной передачи USB подключены. Система может начинать передачу на USB только после того, как число пакетов, соответствующее значению в счетчике переданных пакетов USB для USB-транзакций (импульсов) будет полностью загружено в соответствующий TXFIFO. Это состояние валидно как в режиме “Host”, так и в режиме “Device”. Оно используется только для сверхскоростного (SuperSpeed) режима. Контролепригодность: неограниченно.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
[30]	reserved_30	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0x1. Ограничение записи: writeAsRead.	0x0	R
[31]	reserved_31	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0x1. Ограничение записи: writeAsRead.	0x0	R

26.3.4 Глобальный регистр настроек пороговых значений Rx (GRXTHRCFG)

В стандартном случае импульс Tx начинается, как только в предварительную выборку попадает один пакет; импульс Rx начинается, как только освобождается доступное место для одного пакета. В таких условиях всё работает хорошо до тех пор, пока системная шина работает быстрее, чем шина USB 3.0 (передача пакета весом 1024 байта занимает ~2.2 мкс на шине USB в сверхскоростном (SS) режиме).

Если запаздывание системной шины в процессе получения пакета весом 1024 байта превышает 2.2 мкс, тогда запуск импульса при одно-пакетном условии ведет к раннему прерыванию импульса, что провоцирует нежелательное снижение производительности.

Чтобы избежать недозаполнения и переполнения в процессе генерации импульса, в системной шине с большой задержкой (как у USB) управление размером импульса и пороговыми значениями предоставляется через регистры GTXTHRCFG и GRXTHRCFG. [29] бит регистров GTXTHRCFG и GRXTHRCFG подключает эту функцию.

Более подробно применение этого регистра описано в главе «Детали архитектуры».

Способы выбора значений для полей этого регистра описаны в Разделе «Размеры буферов данных FIFO TX/RX и настройки регистров управления пороговыми значениями TX/RX» в Руководстве пользователя контроллера сверхскоростной USB 3.0 DWC.

Замечания:

- регистр GRXTHRCFG неприменим для целевой отладки (Debug Target).
- существует проблема, когда в процессе импульса передачи типа bulk OUT контроллер устройства производит отправку пакета ACK с NumP=0, за которым следует пакет ACK с NumP=1 без пакета ERDY. Контроллеры хоста USB 3.0 от стороннего изготовителя это может спровоцировать продолжить ожидание прихода пакета ERDY.

В спецификации USB 3.0 утверждается: «когда конечная точка не находится в условиях внешнего управления потоком (flow control condition), она не будет отсылать пакет ERDY,

если только это не конечная точка типа Bulk, которые поддерживают работу с потоком (streams)». В данном случае после того, как устройство произвело отправку пакета ACK (nump=1), конечная точка не находилась в условиях передачи данных, контролируемой со стороны хоста (flow control), так что она не отослала пакет ERDY.

Устройство отослало бы пакет ERDY, если бы следующий пакет OUT не был получен. В момент, когда следующий пакет OUT был получен, имелось достаточно буферного пространства, чтобы принять его, так что устройство приняло пакет, проинформировав хост, что пакет больше не находится в потоке передачи данных. Хост должен дождаться возвращения ответов для каждого из пакетов OUT и затем принять решение о том, находится ли еще конечная точка в условиях передачи данных, контролируемой со стороны хоста или нет.

Спецификация USB 3.1 замещает всю спецификацию USB 3.0. В корректировочном списке (errata) утверждается: «если хост продолжает или возобновляет транзакции к конечной точке, конечная точка произведет переоценку своего статуса управления относительно потока передачи данных и даст соответствующий ответ». Однако в USB 3.0 нет заметок по инженерным исправлениям (ECN) данной проблемы.

Для обхода этой проблемы, необходимо отключить глобальный режим пороговых значений Rx (Global Rx Threshold mode) путем установки GRXTHRCFG.UsbRxPktCntSel=0. Вместо этого, для предотвращения отправки устройством пакета ACK с NumP=0 можно также программно задать DCFG.NUMP режим (в котором всегда передается фиксированный пакет NUMP) вместо режима для nump, основанного на пороговых значениях RX (RX threshold based nump mode). В пакете ACK, NUMP – это минимальное значение (DCFG.NUMP, bMaxBurstSize) для каждой конечной точки.

Поля регистра GRXTHRCFG описаны в Таблица 26.14.

Таблица 26.14. Глобальный регистр настроек пороговых значений Rx

Диапазон	Название	Описание	Исходное значение	Тип доступа
[12:0]	ResvISOCOUTSpc	Место, зарезервированное в буфере Rx FIFO для изохронных (ISOC) пакетов OUT (Space reserved in Rx FIFO for ISOC OUT). В режиме “Host” это поле неприменимо и в него должен быть запрограммирован «0». В режиме “Device” это значение отражает объем памяти, который следует зарезервировать для изохронных (ISOC) пакетов OUT. Программируемое значение следует выбирать таким образом, чтобы неизохронные пакеты не были полностью потеряны (dropped). Если не нужно резервировать место под изохронные пакеты OUT, следует запрограммировать в это поле «0».	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Это поле валидно только в режиме “Device”. Максимальный конфигурируемый объем буфера RX FIFO – 8192. Таким образом, это поле имеет ширину 13 битов. Величина зарезервированного пространства записана в единицах DWC_USB3_MDWIDTH. Для сверхскоростного режима (SS), объем резервируемого пространства всегда является округлением граничных объемов ближайшего пакета. Таким образом, рекомендуется всегда программировать сюда значение, соответствующее MPS (максимальный размер пакета) или числам, кратным ему. Для высокоскоростного/полноскоростного режимов (HS/FS) объем резервируемого пространства - это имеющееся значение. Замечание: для сверхскоростного режима (SS) следует резервировать место для изохронных (ISOC) пакетов в том случае, когда буфер Rx FIFO может вместить 2 или более MPS. В противном случае, это может привести к снижению производительности для неизохронных пакетов. Если пространство полностью занято под изохронные пакеты, неизохронные будут полностью потеряны. Для того, чтобы решить этот вопрос в процессе конфигурирования системы, следует обратиться к Разделу «Пути получения данных в режиме “Device”» (“Device-Mode Receive Path”) в главе 3 справочника по сверхскоростному контроллеру USB 3.0 DWC.		
[14:13]	reserved_14_13	Зарезервировано. Контролепригодность: writeAsRead. Макса сброса: 0x3. Ограничение записи: writeAsRead.	0x0	R
[15]	reserved_15	Зарезервировано. Маска сброса: 0x1.	0x0	R
[18:16]	reserved_18_16	Зарезервировано. Контролепригодность: writeAsRead. Макса сброса: 0x7. Ограничение записи: writeAsRead.	0x0	R
[23:19]	UsbMaxRxBurstSize	Максимальный размер получаемого импульса в USB (USB Maximum Receive Burst Size). В режиме “Host” это поле определяет максимальный импульс Bulk IN, который система DWC_usb3 может произвести. Когда системная шина медленнее, чем шина USB, буфер RX FIFO может переполниться в процессе длинного импульса.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Можно запрограммировать в это поле меньшее значение, чтобы ограничить размер импульса RX, который может произвести система. Это применимо только к конечным точкам типов SS Bulk, Interrupt IN и Isochronous, находящимся в режиме “Host”. В режиме “Device” это поле определяет значение NUMP, которое посылается в пакете ERDY конечной точке OUT. Запрограммированное значение не должно превышать размер буфера RX FIFO. Это поле валидно только когда значение UsbRxPktCntSel = 1. Валидные значения этого поля от 1 до 16.		
[27:24]	UsbRxPktCnt	Счетчик полученных пакетов в USB (USB Receive Packet Count). В режиме “Host” это поле определяет пространство (измеряемое количеством пакетов), которое должно быть доступно в буфере RX FIFO перед тем, как система может начать генерировать соответствующую транзакцию USB RX (импульс). В режиме “Device” это поле задает пространство (измеряемое количеством пакетов), которое должно быть доступно в буфере RX FIFO перед тем, как система может отправить пакет ERDY конечной точке, находящейся в условиях передачи данных, контролируемой со стороны хоста (flow-controlled endpoint). Это поле валидно только когда в поле подключения счетчика полученных пакетов USB установлено значение «1». Валидные значения этого поля от 1 до 15. Замечание: значение этого поля должно быть меньше или равно значению поля максимального размера получаемого импульса в USB (UsbMaxRxBurstSize).	0x0	R/W
[28]	reserved_28	Зарезервировано. Контролепригодность: writeAsRead. Макса сброса: 0x1. Ограничение записи: writeAsRead.	0x0	R
[29]	UsbRxPktCntSel	Подключение счетчика полученных пакетов USB (USB Receive Packet Count Enable). Это поле подключает/отключает ограничения многопакетного приема USB: 0: система может начинать получение на USB только когда в буфере RX FIFO есть место хотя бы под один пакет; 1: система может начинать получение на USB только когда в буфере RX FIFO есть место хотя бы под количество пакетов, определяемое значением UsbRxPktCnt. Этот вариант валиден как в режиме “Host”, так и в режиме “Device”. Он используется только в сверхскоростном (SS) режиме.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		В режиме “Device”: - установка «1» в этот бит также подключает функционал отправки отчетов NUMP в пакетах ACK на основе пространства, имеющегося в буфере RX FIFO, вместо отправки отчетов с фиксированной величиной NUMP, вычисленной из значения DCFG.NUMP для не-контрольных конечных точек. - если используется функция внешнего контроля буфера (EBC), следует отключить этот режим путем установки UsbRxPktCntSel = 0.		
[31:30]	reserved_31_30	Зарезервировано. Контролепригодность: writeAsRead. Макса сброса: 0x3. Ограничение записи: writeAsRead.	0x0	R

26.3.5 Глобальный регистр системных настроек (GCTL)

Подробнее о DWC_USB3_GCTL_INIT написано в <workspace>/src/DWC_usb3_params.v.

Замечание: когда гибернация не подключена, в GblHibernationEn можно писать любое значение. Оно всегда возвращает 0 при чтении.

Поля регистра GCTL описаны в Таблица 26.15.

Таблица 26.15. Глобальный регистр системных настроек

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	DSBLCLKGTNG	Отключение стробирования тактовых импульсов (Disable Clock Gating), DsblClkGtng. Когда в этот бит установлено значение «1» и система находится в режиме низкого питания, внутреннее стробирование тактовых импульсов отключено. После сброса при запуске возможна установка в этот бит значения 1'b1.	0x0	R/W
[1]	GblHibernationEn	Этот бит подключает гибернацию на глобальном уровне. Если гибернация не подключена через этот бит, PMU (Power Management Unit - модуль управления питанием) немедленно принимает запросы на изменение режима питания D0->D3 и D3->D0, но не сохраняет и не восстанавливает никакие системные состояния. К тому же, модули PMU никогда не управляют интерфейсами PHY и предоставляют продолжать управление интерфейсами PHY системе. Ограничение записи: writeAsRead.	0x0	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
[2]	U2EXIT_LFPS	Если в этот бит установлено значение: 0: канал интерпретирует сигнал LFPS как валидный выход из режима U2; 1: канал ожидает 8 мкс при получении сигнала LFPS перед обнаружением валидного выхода из режима U2. Этот бит добавлен для улучшения совместимости со сторонним контроллером хоста/устройства. Контроллер хоста/устройства в режиме U2, находясь в процессе обнаружения приемника (receiver detection), генерирует сбой LFPS длительностью около 4 мс. Это провоцирует хост/устройство выйти из режима U2, потому что значение LFPS-фильтра – 248 нс. С подключением нового функционала, хост/устройство может оставаться в режиме U2, игнорируя сбой, генерируемый контроллером хоста/устройства. Этот бит применим как для контроллера хоста, так и для контроллера устройства.	0x1	R/W
[3]	DISSCRAMBLE	Отключение кодирования (Disable Scrambling) DisScramble. Запрос на передачу к партнеру канала (Link Partner) на следующем переключении к состоянию восстановления (Recovery) или упорядочивания (Polling).	0x0	R/W
[5:4]	SCALEDOWN	Режим понижения масштаба (Scale-Down Mode) ScaleDown. Когда режим понижения масштаба подключен для симуляции, система использует пропорционально уменьшенные величины задержек, в результате симуляции работают быстрее. Когда режим понижения масштаба отключен, используются текущие значения задержек. Это требуется для аппаратной работы. Высокоскоростной/полноскоростной/низкоскоростной (HS/FS/LS) режимы: - 2'b00: отключает все понижения масштаба. Используются текущие значения задержек. - 2'b01: подключает пропорциональное уменьшение всех задержек, кроме задержек приостановки и возобновления режима “Device”. Это включает и вычисление скорости (Speed enumeration), и протоколы HNP/SRP, и приостановку и возобновление режима “Host”. - 2'b10: подключает только пропорциональное уменьшение задержек приостановки и возобновления режима “Device”. - 2'b11: подключает пропорциональное уменьшение задержек 0-го и 1-го бита. Сверхскоростной (SS) режим: - 2'b00: отключает все понижения масштаба. Используются текущие значения задержек.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		- 2'b01: подключает пропорциональное уменьшение задержек SS-режима и повторных значений, включая: (1) уменьшение количества тренинговых последовательностей TxEq (TxEq training sequences) до 8; (2) уменьшение времени упорядочивающего импульса LFPS (LFPS polling burst) до 256 нс; (3) уменьшение времени получения сигнала теплого сброса LFPS (LFPS warm reset) до 30 мкс. Полный список задержек указан в файле rtl_vip_scaledown_mapping.xls в директории <workspace>/sim/SoC_sim. - 2'b10: отправка тренинговых последовательностей TxEq (TxEq training sequences) не производится. Такой режим замещает бит 4. - 2'b11: подключает пропорциональное уменьшение задержек 0-го и 1-го бита.		
[7:6]	RAMCLKSEL	Выбор синхросигнала RAM (RAM Clock Select) RAMClkSel. - 2'b00: синхросигнал шины (bus_clk); - 2'b01: синхросигнал канала (pipe_clk) (используется только в режиме “Device”); - 2'b10: в режиме “Device”: синхросигнал канала, делённый пополам (pipe/2 clock). В режиме “Host”: контроллер переключает ram_clk между синхросигналами pipe/2 clock, mac2_clk и bus_clk в зависимости от состояния портов U2/U3. - 2'b11: в режиме “Device” задает mac2_clk в качестве ram_clk (когда используется 8-битный UTMI или ULPI. Не поддерживается в режиме 16-битного UTMI). В режиме “Host”, контроллер переключает ram_clk между синхросигналами pipe_clk, mac2_clk и bus_clk в зависимости от состояния портов U2/U3. В режиме “Device”, при сбросе USB и при отключении USB, аппаратура производит очистку этих битов к значению 2'b00. Более подробно способ выбора синхросигнала RAM описан в Разделе “Требования к генерации синхросигнала и синтезу дерева синхронизации (CTS)” в главе «Детали архитектуры» в справочнике по сверхскоростному контроллеру USB 3.0 систем DesignWare. Замечания: - в режиме “Device”, если установить значение 2'b11 (mac2_clk) в RAMClkSel, контроллер внутренне переключает синхросигнал ram_clk на bus_clk, когда состояние канала изменяется на режим возобновления или U2. - в режиме “Host”, если выбрано значение 2/3, тогда контроллер переключает ram_clk между синхросигналами bus_clk, mac2_clk и pipe_clk, pipe_clk/2 на основе состояния портов U2/U3.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Например, если активен только порт U2, а порты U3 приостановлены, тогда ram_clk переключается на mac2_clk. Когда активны только порты U3, а порты U2 приостановлены, тогда система внутренне переключает ram_clk на синхросигнал pipe3. Когда приостановлены все порты U2 и U3, она переключает ram_clk на bus_clk. Это позволяет отделить ram_clk от bus_clk и, в зависимости от требований к полосе частот, позволяет запускать работу bus_clk на более низких частотах, чем прописано в требованиях к ram_clk. Тем не менее, частоты bus_clk не могут опускаться ниже 60 МГц в режиме “Host”, и это не верифицировано. Значение «2» может быть выбрано только в том случае, если ширина канала данных составляет 8 или 16 битов. В этом случае, когда ram_clk переключается на pipe_clk, вместо pipe_clk используется pipe_clk/2. Если для RAMClkSel выбрано значение «3», значит, когда ram_clk переключается на pipe_clk, используется именно pipe_clk безо всякого делителя. - в режиме “Device”, когда RAMClkSel != 2'b00, частота bus_clk_early может принимать минимальное значение 1 МГц. Это протестировано в симуляции, а также на аппаратуре с драйверами хоста систем Linux, Microsoft Windows 8 и MCCI Windows7. Когда частота bus_clk составляет 1 МГц, поддерживаются только контрольные (control) и непериодические (non-periodic) передачи. Для периодических приложений, минимальная частота bus_clk_early будет выше, в зависимости от приложения и шины SoC (Systems-On-Chip). Даже несмотря на то, что частота в 1 МГц протестирована на стандартных драйверах хоста, Synopsys рекомендует для реализаций ASIC минимальную частоту 5 МГц, чтобы обеспечить резерв или хотя бы запасную возможность увеличения частоты bus_clk до 5 МГц, если понадобится. Программирование случайных данных в это поле провоцирует побочные эффекты. Не рекомендуется применять на этом регистре тестирование Bit Bash. Контролепригодность: writeAsRead.		

Диапазон	Название	Описание	Исходное значение	Тип доступа
[8]	DEBUGATTACH	Присоединение при отладке (Debug Attach). Когда этот бит установлен: - сверхскоростной (SS) канал переходит напрямую в состояние упорядочивания канала (Polling link state) (после установки RUN/STOP в регистре DCTL) без проверки удаленной терминации; - таймаут упорядочивания канала LFPS (Link LFPS polling timeout) бесконечен; - таймаут упорядочивания в процессе TS1 бесконечен (в том случае, если канал ожидает завершения TXEQ).	0x0	R/W
[9]	U1U2TimerScale	Отключает уменьшение длительности таймера U1/U2 (Disable U1/U2 timer Scaledown), U1U2TimerScale. Если в этот бит установлено значение «1» и GCTL[5:4] (ScaleDown) = 2'bX1, это отключает пропорциональное уменьшение значений неактивного таймера U1/U2. Предназначено только для режима симуляции.	0x0	R/W
[10]	SOFITPSYNC	Если в этот бит установлено значение «0» при работе в режиме “Host”, система удерживает UTMI/ULPI PHY на первом порте в не-приостановленном состоянии всегда, когда присутствует сверхскоростной (SS) порт, который не находится в Rx.Detect, SS.Disable и U3. Если в этот бит установлена «1» при работе в режиме “Host”, система удерживает UTMI/ULPI PHY на первом порте в не-приостановленном состоянии в те моменты, когда другие не-сверхскоростные порты не находятся в приостановленном состоянии. Эта функция полезна потому, что приостановка UTMI/ULPI , когда активен только сверхскоростной порт, сберегает энергию. Также это помогает разрешить проблему, когда PHY не производит передачу возобновлния хоста (host resume) до тех пор, пока не будет помещена в состояние приостановки. Этот бит должен быть запрограммирован как часть процесса инициализации при сбросе во время запуска (power-on reset), и не должен быть динамически изменен впоследствии. Замечания: - USB2PHYCFGn[6].PhySusp в конце концов принимает решение о помещении UTMI/ULPI PHY в состояние приостановки. К тому же, когда в этот бит установлено значение «1», система генерирует пакеты ITP по счетчику на основе ref_clk. В противном случае, пакеты ITP и SOF генерируются по счетчику на основе utmi/ulpi_clk[0]. Для того, чтобы запрограммировать период опорного синхросигнала (reference clock period) внутри системы, следует обращаться к GUCTL[31:22].REFCLKPER.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		- эта функция валидна в конфигурациях “Host” и DRD/OTG, и используется только при работе в режиме “Host”. - если эта функция или GFLADJ.GFLADJ_REFCLK_LPM_SEL никогда не будут использованы, минимальную частоту ref_clk можно снизить вплоть до 32 КГц. Можно подсоединить suspend_clk (не выше 32 КГц) к ref_clk. - если планируется подключение аппаратного или программного (PORTPMSC. HLE=1) управления питанием каналов (LPM), эту функцию нельзя использовать. Следует отключить её, установив «0» в этот бит и использовать функцию GFLADJ.GFLADJ_REFCLK_LPM_SEL. - если в этот бит установлено значение «1», следует обнулить бит GUSB2PHYCFG.U2_FREECLK_EXISTS и параметр DWC_USB3_FREECLK_USB2_EXIST. Если планируется работа системы в режиме USB 3.0, следует запрограммировать в этот бит значение «0».		
[11]	CORESOFTRESET	Мягкий сброс системы (Core Soft Reset), CoreSoftReset. 1'b0 - мягкого сброса нет; 1'b1 - мягкий сброс системы. Производит очистку прерываний и всех регистров CSR, кроме следующих: - GCTL; - GUCTL; - GSTS; - GSNPSID; - GGPIO; - GUID; - регистры GUSB2PHYCFGn; - регистры GUSB3PIPECTLn; - DCFG; - DCTL; - DEVTEN; - DSTS. Когда произведен сброс PHY (через регистры GUBS3PHYCFG или GUSB3PIPECTL), необходимо удерживать систему в состоянии сброса до тех пор, пока синхросигналы PHY не стабилизируются. Это касается сбросов шины, памяти ram и домена mac. Более подробно это описано в Разделе «Сброс генерации» в главе «Детали архитектуры» в справочнике по сверхскоростному контроллеру USB 3.0 систем DesignWare. Замечание: этот бит применяется только в целях отладки. В режиме xHCI для мягкого сброса	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		следует пользоваться USBCMD.HCRESET, а в режиме “Device” – DCTL.SoftReset. Программирование случайных данных в это поле сбросит внутреннюю логику контроллера хоста. В связи с этим побочным эффектом применять к этому регистру тестирование Bit Bash не рекомендуется. Контролепригодность: writeAsRead.		
[13:12]	PRTCAPDIR	Направление возможностей порта (Port Capability Direction), PrtCapDir. - 2'b01: для конфигураций хоста; - 2'b10: для конфигураций устройства; - 2'b11: для конфигураций OTG. Когда значение DWC_USB3_EN_OTG = 0, система ведет себя как DRD (устройство двойного назначения, Dual Role Device). Когда значение DWC_USB3_EN_OTG = 1: - если PrtCapDir = 2'b11, система ведет себя как устройство OTG 2.0 с A-устройством или B-устройством (A-device or B-device), что определяется вводом IDDIG, , и ролью “Host” или “Peripheral” на основе протокола HNP. - если PrtCapDir = 2'b01, система ведет себя как устройство DRD в режиме “Host”. - если PrtCapDir = 2'b10, система ведет себя как устройство DRD в режиме “Device”. Устройство OTG может быть запрограммированно на подключение/отключение протоколов SRP и HNP за счёт использования полей, присутствующих в регистре OCFG. Когда значение DWC_USB3_EN_OTG = 2: - если PrtCapDir = 2'b11, система ведет себя как устройство OTG 3.0 с A-устройством или B-устройством (A-device or B-device), что определяется вводом IDDIG, и ролью “Host” или “Peripheral” на основе протокола HNP/RSP. - если PrtCapDir = 2'b01, система ведет себя как DRD в режиме “Host”. В этом режиме, сигнал VBUS устанавливается сразу после сброса при запуске. Если контроллер DWC_usb3 будет подключен к PC и, позднее, приложение сконфигурирует устройство, сигнал VBUS будет установлен на USB с обеих сторон кабеля. Это может потенциально привести к повреждениям PC как принимающего устройства. - Если PrtCapDir = 2'b10, система ведет себя как DRD в режиме “Device”. Устройство OTG может быть запрограммированно на подключение/отключение протоколов SRP и HNP/RSP за счёт использования полей, присутствующих в регистре OCFG. Замечание: для статических приложений, работающих только в режиме “Host”/”Device”,	0x2	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		следует использовать DRD режим “Host” или DRD режим “Device”. Для таких приложений не рекомендуется комбинация GCTL.PrtCapDir=2'b11 с отключением протоколов SRP и HNP/RSP. Последовательность для переключения режимов в конфигурациях DRD следующая. Переключение из режима “Device” в режим “Host”: 1. Сброс контроллера через GCTL[11] (CoreSoftReset); 2. Установка GCTL[13:12] (PrtCapDir) = 2'b01 (режим “Host”); 3. Сброс хоста через USBCMD.HCRESET; 4. Далее следует выполнить последовательность шагов, изложенную в Разделе «Инициализация регистров хоста» справочника по сверхскоростному контроллеру USB 3.0 DWC. Переключение из режима “Host” в режим “Device”: 1. Сброс контроллера через GCTL[11] (CoreSoftReset); 2. Установка GCTL[13:12] (PrtCapDir) = 2'b10 (режим “Device”). 3. Сброс прибора через DCTL[30] (CSftRst). 4. Далее следует выполнить последовательность шагов, изложенную в Разделе «Инициализация регистра» справочника по сверхскоростному контроллеру USB 3.0 DWC. Программирование случайных значений в это поле заставляет систему переключаться между режимом “Host” и режимом “Device”. Не рекомендуется применять на этом регистре тестирование Bit Bash. Контролепригодность: writeAsRead.		
[15:14]	FRMSCLDWN	Это поле пропорционально уменьшает длительность SOF/USOF/ITP для устройства. Для сверхскоростного/высокоскоростного (SS/HS) режимов: - значение 2'h3 задает интервал 15.625 мкс; - значение 2'h2 задает интервал 31.25 мкс; - значение 2'h1 задает интервал 62.5 мкс; - значение 2'h0 задает интервал 125 мкс; Для полноскоростного (FS) режима, масштабируемое значение умножается на 8. Когда подключена возможность отладки xHCI (Debug Capability), это поле так же пропорционально уменьшает MaxPacketSize конечных точек типов bulk IN и OUT, чтобы позволить больший трафик в процессе	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		симуляции. Во время симуляции оно может быть изменено только из ненулевого значения. - 2'h0: 1024 байта; - 2'h1: 512 байт; - 2'h2: 256 байт; - 2'h3: 128 байт.		
[16]	U2RSTECN	Если происходит сбой сверхскоростного (SS) подключения в процессе обмена POLL или LMP, устройство подключается в не-сверхскоростном режиме. Если этот бит установлен, тогда устройство еще трижды предпринимает попытку подсоединиться в сверхскоростном режиме, даже если до этого работа в сверхскоростном режиме закончилась сбоем. При каждой попытке устройство проверяет завершенность работы (termination) приемника 8 раз. Начиная с реализации 2.60a, от этого бита зависит, следует ли проверять Rx.Detect 8 раз или 1 раз при каждой попытке. Контроллер устройства при сбросе USB 2.0 производит проверку завершенности работы приемника 8 раз за попытку, если в этот бит установлен «0», или только один раз за попытку, если в бит установлена «1». Замечание: этот бит применим только в режиме “Device”.	0x1	R/W
[17]	BYPSSETADDR	Обход SetAddress в режиме “Device” (Bypass SetAddress in Device Mode). Когда установлен бит BYPSSETADDR, система устройства напрямую использует значения битов DCFG[DevAddr] для сравнения адреса устройства в токенах. При симуляции эту функцию можно использовать для того, чтобы избежать отправки настоящей контрольной передачи SET ADDRESS на USB, и для того, чтобы заставить систему устройства ответить на новый адрес. Когда возможность отладки xHCI подключена и этот бит установлен, целевой объект отладки немедленно входит в сконфигурированное состояние, не требуя от отладочного хоста отправлять запрос SetAddress или SetConfig. Замечание: этот бит можно устаналвивать только для симуляции. В настоящей аппаратуре в этот бит следует устанавливать 1'b0.	0x0	R/W
[18]	MASTERFILTBYPASS	Основной обход фильтров (Master Filter Bypass). Когда в этот бит установлено 1'b1, то, вне зависимости от выбранного параметра `DWC_USB3_EN_BUS_FILTERS, производится обход всех фильтров в модуле DWC_usb3_filter. Производится также обход двойных синхронизаторов mac_clk, предшествующих фильтрам. Для подключения фильтров в этот бит должно быть установлено 1'b0.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
[31:19]	PWRDNSCALE	Масштабируемое понижение питания (Power Down Scale), PwrDnScale. Ввод USB3 suspend_clk замещает pipe3_rx_pclk в качестве источника синхроимпульсов для небольшой части системы USB3, которая запускается, когда SS PHY находится в самом низком режиме питания (P3) и, таким образом, не генерирует синхросигнал. Поле масштабируемого понижения питания задает число периодов suspend_clk, которые помещаются в период синхросигнала на частоте 16 КГц. При делении остаток следует округлить. Например, при использовании 8-битной/16-битной/32-битной PHY и синхросигнала приостановки (Suspend clock) на частоте 25 МГц, значение поля масштабируемого понижения питания: 25000 КГц/16 КГц = 13'd1563 (округляя вверх). Замечание: - минимальная частота приостановки синхросигнала - 32 КГц; - максимальная частота приостановки синхросигнала - 125 МГц. LTSSM во время режима приостановки использует синхросигнал приостановки для таймеров по 12 мс и 100 мс. В соответствии со спецификацией USB 3.0, точность на этих таймерах - от 0% до +50%. 12 мс + 0~+50% точности = 18 мс (диапазон 12 мс - 18 мс); 100 мс + 0~+50% точности = 150 мс (диапазон 100 мс - 150 мс). Требования к точности синхросигнала приостановки: (12,000/62.5) * (GCTL[31:19]) * текущий suspend_clk_period должен быть от 12,000 до 18,000; (100,0000/62.5) * (GCTL[31:19]) * текущий suspend_clk_period должен быть от 100,000 до 150,000; Например, если частота suspend_clk варьируется от 7.5 МГц до 10.5 МГц, тогда величина, которую следует запрограммировать: Power Down Scale = 10500/16 = 657 (округлена вверх; использована максимальная частота).	0x618	R/W

26.3.6 Глобальный регистр статуса управления питанием (GPMSTS)

Этот регистр отладки даёт информацию о том, какое событие спровоцировало выход из состояния гибернации. Он содержит внутренний статус и информацию по машине состояний (state machine information), и предназначен для Synopsys, исключительно в целях отладки.

В режиме, использующем только USB 2.0, этот регистр неприменим.

Поля регистра GPMSTS описаны в Таблица 26.16.

Таблица 26.16. Глобальный регистр статуса управления питанием

Диапазон	Название	Описание	Исходное значение	Тип доступа
[9:0]	U2Wakeup	Указывает следующие условия пробуждения порта USB 2.0: бит [0]: обнаружен ток перегрузки; бит [1]: обнаружено возобновление; бит [2]: обнаружено соединение; бит [3]: обнаружено отсоединение; бит [4]: последнее состояние соединения; бит [5]: обнаружена смена идентификатора; бит [6]: обнаружен запрос SRP; бит [7]: обнаружено прерывание ULPI; бит [8]: обнаружен сброс USB; бит [9]: обнаруженное возобновление изменено. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R
[11:10]	reserved_10_11	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0x3. Ограничение записит: writeAsRead.	0x0	R
[16:12]	U3Wakeup	Дает следующие условия пробуждения порта USB 3.0: бит [12]: обнаружен ток перегрузки; бит [13]: обнаружено возобновление; бит [14]: обнаружено соединение; бит [15]: обнаружено отсоединение; бит [16]: последнее состояние соединения. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R
[27:17]	reserved_27_17	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0x7ff. Ограничение записит: writeAsRead.	0x0	R
[31:28]	PortSel	Выбирает номер порта. Контролепригодность: нетестируемо.	0x0	W

26.3.7 Глобальный регистр статуса (GSTS)

Поля регистра GSTS описаны в Таблица 26.17.

Таблица 26.17. Глобальный регистр статуса

Диапазон	Название	Описание	Исходное значение	Тип доступа
[1:0]	CURMOD	Текущий режим работы (Current Mode of Operation). Указывает текущий режим работы: 2'b00: режим “Device”; 2'b01: режим “Host”. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R
[3:2]	reserved_3_2	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0x3. Ограничение записи: writeAsRead.	0x0	R
[4]	BUSERRADDRVLD	Валидный адрес ошибки шины (Bus Error Address Valid). Указывает, что регистр GBUSERRADDR валиден, и дает отчет о первом адресе шины, на котором встречается ошибка шины. Замечание: поддерживается только в конфигурациях AHB и AXI. Маска сброса: 0x1. Изменчивость: true.	0x0	R/W
[5]	CSRTimeout	Таймаут CSR (CSR Timeout). Когда этот бит имеет значение 1'b1, он указывает, что программное обеспечение произвело запись или чтение из регистра системы, которое не вышло завершить в пределах `DWC_USB3_CSR_ACCESS_TIMEOUT тактов синхросигнала шины (по умолчанию: h1FFFF). Маска сброса: 0x1. Изменчивость: true.	0x0	R/W
[6]	Device_IP	Прерывание устройства в очереди на обработку (Device Interrupt Pending). Это поле указывает, что присутствует прерывание, ожидающее обработки, которое относится к работе периферийного устройства в очереди событий “Device”. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
[7]	Host_IP	Прерывание хоста в очереди на обработку (Host Interrupt Pending). Это поле указывает, что присутствует прерывание, ожидающее обработки, которое относится к работе xHC в очереди событий “Host”. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R
[8]	ADP_IP	Прерывание протокола ADP в очереди на обработку (ADP Interrupt Pending) (Attach Detection Protocol - протокол обнаружения подсоединения). Это поле указывает, что присутствует прерывание, ожидающее обработки, которое относится к ADP в регистре ADPEVT. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R
[9]	BC_IP	Прерывание зарядного устройства в очереди на обработку (Battery Charger Interrupt Pending). Это поле указывает, что присутствует прерывание, ожидающее обработки, которое относится к BC (Battery Charger - зарядное устройство) в регистре BCEVT. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R
[10]	OTG_IP	Прерывание OTG (On-The-Go – устройство «на ходу») в очереди на обработку (OTG Interrupt Pending). Это поле указывает, что присутствует прерывание, ожидающее обработки, которое относится к OTG в регистре OEVT. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R
[11]	SSIC_IP	Прерывание SSIC (SuperSpeed InterChip) в очереди на обработку (SSIC interrupt pending). Это поле указывает, что присутствует прерывание, ожидающее обработки, которое относится к SSIC в регистре SEVT. Замечание: когда в параметр DWC_USB3_NUM_SSIC_PORTS установлен 0, этот бит зарезервирован. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
[19:12]	reserved_19_12	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0xff. Ограничение записи: writeAsRead.	0x0	R
[31:20]	CBELT	Текущее значение BELT (Current BELT Value). В режиме “Host” это поле указывает минимальное из всех полученных значений BELT устройства, и значение BELT, которое задано командой установки допуска запаздывания (Set Latency Tolerance Value). Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x7e8	R

26.3.8 Глобальный регистр настроек пользователя 1 (GUCTL1)

Поля регистра GUCTL1 описаны в Таблица 26.18.

Таблица 26.18. Глобальный регистр настроек пользователя 1

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	LOA_FILTER_EN	Если этот бит установлен, то проверка помех порта USB 2.0 будет произведена, как минимум, три раза подряд, прежде чем порт будет отключен. Это предотвращает ложное срабатывание условия против помех (babble condition) при использовании низкокачественных кабелей. Замечание: этот бит валиден только в режиме “Host”.	0x0	R/W
[1]	OVRLD_L1_SUSP_COM	Если этот бит установлен, utmi_l1_suspend_com_n будет перегружен сигналом utmi_sleep_n. Этот бит обычно устанавливается, если PHY отключает синхросигнал порта во время условий сна L1. Замечание: относительно этого бита рекомендуется соединение сигналов SUSPENDM/SLEEPM к PHY, описанное ниже. Для ненулевых портов: следует соединять: - utmi_sleep_n[n] с SLEEPM[n]; - (utmi_suspend_n[n] и utmi_l1_suspend_n[n]) с SUSPENDM[n]; - USB2 PHYCLK[n] с utmi_clk[n]. GUCTL1.OVRLD_L1_SUSP_COM влияет только на Port0.	0x1	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Для Port0: (для PHY, разработанной Synopsys): GUSB2PHYCFGn.U2_FREECLK_EXISTS=1; с этим соединением, PHY поддерживает активную ФАПЧ (PLL) так, что сигнал FREECLK доступен всегда, вне зависимости от нахождения в режиме приостановки или сна. Следует соединять: - USB2 PHY COMMONONN с 0; - utmi_sleep_n[0] с SLEEPM[0]; - (utmi_suspend_n[0] и utmi_l1_suspend_n[0]) с SUSPENDM[0]; - USB2 PHY FREECLK с utmi_clk[0]: - оставить utmi_suspend_com_n, utmi_l1_suspend_com_n неподсоединенными; - в GUCTL1.OVRLD_L1_SUSP_COM может быть установлено любое значение. (Для PHY, разработанной сторонними компаниями): GUSB2PHYCFGn.U2_FREECLK_EXISTS=0; с этим соединением PHY, при достижении требуемых условий (например, GUSB2PHYCFGn[8,6], GUCTL1[1], GFLADJ[23], GCTL[10], условие приостановки, подключение HW LPM и т. д.), может выключить все синхросигналы. Следует соединять: - -utmi_suspend_com_n с SUSPENDM[0] (или его эквивалентом); - -utmi_l1_suspend_com_n с SLEEPM[0] (или его эквивалентом); - PHYCLK0 (синхросигнал первого порта) с utmi_clk[0]; - оставить utmi_suspend_n[0], utmi_l1_suspend_ n[0], utmi_sleep_n[0] неподсоединенными; - установить значение 1'b1 в GUCTL1.OVRLD_L1_SUSP_COM.		
[2]	HC_PARCHK_DISABLE	Отключение проверки параметров хоста (HParChkDisable). Когда в этот бит установлено значение «0» (по умолчанию), xHCI проверяет, соответствуют ли поля контекста (context fields) входного слота/конечной точки спецификации xHCI. При обнаружении ошибки параметра в процессе исполнения команды, xHCI генерирует событие TRB с кодом завершения, в котором сообщается об ошибке параметра: “PARAMETER ERROR”. Когда в этот бит установлено значение «1», xHCI не производит проверок параметров и не генерирует код завершения “PARAMETER ERROR”.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
[3]	HC_ERRATA_ENABLE	Подключение ELD хоста (HELDEn). Когда в этот бит установлено значение «1», он подключает поддержку дельты запаздывания выхода (Exit Latency Delta), определенной в корректировочном списке (Errata) xHCI 1.0. Этот бит используется только в режиме “Host”. В этом режиме в него должно быть установлено значение «1».	0x1	R/W
[7:4]	L1_SUSP_THRLD_FOR_HOST	Это поле эффективно только когда в бит L1_SUSP_THRLD_EN_FOR_HOST установлено значение 1. Более подробная информация представлена в описании бита L1_SUSP_THRLD_EN_FOR_HOST.	0x8	R/W
[8]	L1_SUSP_THRLD_EN_FOR_HOST	Этот бит используется только в режиме “Host”. Контроллер хоста устанавливает сигналы вывода utmi_l1_suspend_n и utmi_sleep_n (см. таблицу «сигналы интерфейса LPM» (LPM Interface Signals) в справочнике) следующим образом: 1) контроллер устанавливает сигнал utmi_l1_suspend_n для того, чтобы поместить PHY в глубокий режим низкого питания в L1, когда выполняются оба следующих условия: - используемое значение HIRD/BESL больше или равно значению поля L1_SUSP_THRLD_FOR_HOST. - в бит The L1_SUSP_THRLD_EN_FOR_HOST установлено 1'b1; 2) контроллер устанавливает сигнал utmi_sleep_n на L1, когда выполняется одно из следующих условий: - используемое значение HIRD/BESL меньше, чем значение поля L1_SUSP_THRLD_FOR_HOST. - в бит L1_SUSP_THRLD_EN_FOR_HOST установлено 1'b0.	0x1	R/W
[9]	DEV_HS_NYET_BULK_SPR	0: работа по умолчанию, никаких изменений в ответах устройства; 1: функция подключена, короткие высокоскоростные пакеты типа bulk OUT получают ответы NYET. Этот бит применим только в режиме “Device” (и игнорируется в режиме “Host”) и используется для работы с USB 2.0. Если этот бит установлен, система устройства отправляет в ответ на успешно полученный короткий пакет типа bulk OUT пакет NYET вместо ответа ACK. Если NYET будет отправлен после получения короткого пакета, то хост отправит PING перед отправкой следующего пакета OUT, что одновременно улучшит работу и расчистит кэш/буфер на стороне хоста.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Что касается внутренних процессов системы устройства, обработка короткого пакета (SPR=1) требует определенного времени, и в течение этого времени передача данных по USB является контролируемой со стороны хоста (USB is flow controlled). Получая ответ NYET вместо ACK на короткий пакет, хост в высокоскоростном режиме не станет посылать новые данные OUT-DATA без отправки пинга. Этот бит квази-статичен, таким образом, в процессе работы устройства его значение не должно меняться.		
[10]	RESUME_OPMODE_HS_HOST	Этот бит используется только в режиме “Host”, для настройки поведения рабочего режима USB 2.0 при состоянии HS Resume (возобновления высокоскоростного режима). - Когда в этот бит установлено значение «1», рабочий режим UTMI/ULPI будет одновременно с терминациями HS-режима (HS terminations) изменен на «нормальный» после EOR (end of resume, конец возобновления). Эта функция нужна для поддержки определенных устаревших UTMI/ULPI PHYs. - Когда в этот бит установлено значение «0», рабочий режим UTMI/ULPI будет изменен на «нормальный» через 2 мкс после того, как терминации HS-режима (HS terminations) изменятся после EOR. Это - поведение по умолчанию.	0x0	R/W
[14:11]	reserved_14_11	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0xf. Ограничение записи: writeAsRead.	0x0	R
[15]	PARKMODE_DISABLE_FSLS	Этот бит используется только в режиме “Host”. Он применяется только в целях отладки. Когда в этот бит установлено значение «1», все FS/LS (полноскоростные/низкоскоростные) элементы шины в режиме «парковки» (park mode) отключаются.	0x0	R/W
[16]	PARKMODE_DISABLE_HS	Этот бит используется только в режиме “Host”. Когда в этот бит установлено значение «1», все HS (высокоскоростные) элементы шины в режиме «парковки» (park mode) отключаются. Для улучшения работы в режиме «парковки», планировщик xHCI составляет в очередь три запроса для высокоскоростных асинхронных конечных точек, в каждом по 4 пакета, в один микро-фрейм. Но если устройство медленно работает, и если оно отправляет пакеты NAK (not acknowledged – не подтверждено) больше	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		трёх раз, тогда эта очередь перепланируется только в следующем микро-фрейме. Это может снизить производительность медленного устройства еще сильнее. В некоторых высокоскоростных устройствах (таких, как Sandisk Cruzer Blade 4GB VID:1921, PID:21863 и Flex Drive VID:3744, PID:8552), когда запрос IN посылается в пределах 900 нс относительно ACK предыдущего пакета, такие устройства посылают NAK. Во время соединения с такими устройствами, если это необходимо, можно отключить режим «парковки» с помощью программного обеспечения, если наблюдается падение производительности в системе. Когда режим «парковки» отключен, отключается обработка многопакетного режима и вместо этого планировщик запрашивает только один пакет в единицу времени. Это позволяет поместить до 12 NAK в одном микро-кадре и улучшает производительность таких медленных устройств.		
[17]	PARKMODE_DISABLE_SS	Этот бит используется только в режиме “Host”. Он применяется только в целях отладки. Когда в этот бит установлено значение «1», все SS (сверхскоростные) элементы шины в режиме «парковки» (park mode) отключаются.	0x0	R/W
[18]	NAK_PER_ENH_HS	1: подключает повышение производительности для высокоскоростных (HS) асинхронных конечных точек в присутствии пакетов NAK. 0: повышение производительности не применяется. Если присутствует периодическая конечная точка, и если конечная точка типа bulk, которая также активна, получает пакеты NAK от устройства, это может спровоцировать понижение производительности второй высокоскоростной конечной точки типа bulk, которая получает пакеты ACK от устройства. Установка в этот бит значения «1» заставит контроллер хоста запланировать больше транзакций к асинхронным конечным точкам (форматов bulk/control) и, таким образом, улучшит работу конечной точки типа bulk. Этот управляющий бит должен быть подключен только если существующей производительности с установками по умолчанию недостаточно для целей высокоскоростного (HS) режима. Установка этого бита управляет только HS-передачами и требуется только для них.	0x1	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
[19]	NAK_PER_ENH_FS	1: подключает повышение производительности для полноскоростных (FS) асинхронных конечных точек в присутствии пакетов NAK. 0: повышение производительности не применяется. Если присутствует периодическая конечная точка, и если конечная точка типа bulk, которая также активна, получает пакеты NAK от устройства, это может спровоцировать понижение производительности второй полноскоростной конечной точки типа bulk, которая получает пакеты ACK от устройства. Установка в этот бит значения «1» заставит контроллер хоста запланировать больше транзакций к асинхронным конечным точкам (форматов bulk/control) и, таким образом, улучшит работу конечной точки типа bulk. Этот управляющий бит должен быть подключен только если существующей производительности с установками по умолчанию недостаточно для целей полноскоростного (FS) режима. Установка этого бита управляет только FS-передачами и требуется только для них.	0x0	R/W
[20]	DEV_LSP_TAIL_LOCK_DIS	0: работа по умолчанию, подключает приборную логику замыкания lsp (lsp lock logic) для конечного обновления блока запроса передачи (tail TRB update); 1: отключает данное исправление. Это исправление бага для STAR 9000716195, которое влияет на режим CSP для конечных точек OUT в режиме “Device”. Проблема состояла в том, что конечный индекс блока TRB не синхронизуется с обновлением значения Scratchpad bytecount (счетчика байтов блокнотной памяти) в кэш. Если ускоренный запрос приходит в момент между процессом обновления байтового счетчика (bytecount update) только что вызванного блока TRB и обновления записи конечного индекса (tail-index write update) блока TPF, RDP будет работать над неверным конечным индексом и пропустит декрементацию байтового счетчика для нового вызванного блока TRB в процессе ускорения (fast-forwarding process). Это исправление должно присутствовать всегда.	0x0	R/W
[23:21]	IP_GAP_ADD_ON	Этот регистр используется для добавления надстроек по установке межпакетной паузы по умолчанию в USB 2.0 MAC. Программировать ненулевое значение в этот бит следует только в том случае, когда	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		необходимо увеличить установленный по умолчанию обсчёт межпакетной задержки в модуле USB 2.0 MAC DWC_usb3_u2mac.v.		
[24]	DEV_L1_EXIT_BY_HW	0: (поведение по умолчанию) отключает аппаратную логику выхода устройства из режима L1. 1: функция подключена. Этот бит применим только в режиме “Device” (2.0). Это поле позволяет контроллеру устройства посылать сигнал удаленного пробуждения (remote wakeup) для режима L1, если устройство в состоянии L1 уже готово к отправке/приему данных. Если хост ожидает, что устройство пошлет сигнал удаленного пробуждения для того, чтобы возобновить работу после перехода в состояние L1 в режиме контроля со стороны хоста (flow controlled state), тогда можно установить этот бит, чтобы посылать сигнал удаленного пробуждения автоматически, когда контроллер устройства будет готов. Эта аппаратная функция удаленного пробуждения применима только к передачам типов bulk и interrupt, но не к изохронным/управляющим (Isoch/Control). - Когда устройство занято работой с контрольными (control) передачами, сигналы управления питанием каналов (Link Power Management, LPM) будут получать отказы (ответы NYET). Сигналы LPM будут приняты только после завершения контрольных передач (индикатором завершения служат пакеты ACK либо STALL). - Для изохронных (Isoch) передач, хосту необходимо провести пробуждение и начать передачу. Контроллер устройства не будет производить пробуждение, пока готовятся изохронные конечные точки. SW (secure world - изолированная безопасная среда исполнения) прибора обязана удерживать GUSB2PHYCFG[EnblSlpM] в состоянии сброса, чтобы синхросигнал PHY исправно работал и давал возможность отслеживать интервалы начала кадра (SOF). - Когда подключен режим сна (гибернации) L1, контроллер не будет производить автоматический выход для запросов гибернации (hibernation requests) через L1. Этот бит квази-статичен, таким образом, он не должен быть изменен в процессе работы устройства.	0x0	R/W
[25]	P3_IN_U2	0: поведение по умолчанию: когда сверхскоростной канал (SuperSpeed link) находится в режиме U2, в интерфейсе PIPE	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		будет произведена попытка перейти в режим питания (Power State) P2. 1: когда сверхскоростной канал (SuperSpeed link) находится в режиме U2, будет произведена попытка перейти в режим питания P3, если GUSB3PIPECTL[17] установлен. Установка этого бита подключает режим питания P3, когда сверхскоростной канал находится в режиме U2. Это еще один способ сбережения энергопитания. Следует свериться с поставщиком PHY перед подключением этой опции. При установке значения «1» в этот бит для подключения режима питания P3 в P2, необходимо, чтобы GUSB3PIPECTL[27] был обнулен, для того, чтобы выход из U2 производился только в режиме P0. Этот бит должен быть установлен только в том случае, когда GCTL.SOFITPSYNC=1 или GFLADJ.GFLADJ_REFCLK_LPM_SEL=1.		
[26]	DEV_FORCE_20_CLK_FOR_30_CLK	0: поведение по умолчанию: при работе в режиме 2.0 используется синхросигнал 3.0. 1: функция подключена. Этот бит применим (и должен быть установлен) только в режиме “Device” (DCFG.Speed != SS). В версии 3.0 системы устройства, если система запрограммирована работать только в режиме 2.0 (таким образом, скорость устройства (Device Speed) запрограммирована для скоростей режима 2.0 в DCFG[Speed]), то установка этого бита провоцирует перенаправление внутреннего синхросигнала 2.0 (utmi/ulpi) в качестве синхросигнала 3.0 (pipe). Подключение этой функции позволяет синхросигналу pipe3 не работать при принудительной работе в режиме “Device” 2.0. Замечания: - при применении этой функции, все вводы pipe3 должны быть в неактивном состоянии. В особенности не должны работать синхросигналы pipe3. pipe3_phystatus_async должен быть привязан к 0. Этот бит не следует устанавливать, если система запрограммирована на работу в сверхскоростном режиме (SuperSpeed mode) (даже когда она возвращается к режиму 2.0). - этот бит квази-статичен, таким образом, он не должен изменяться в процессе работы. - если подключен параметр “DWC_USB3_REMOVE_PIPE_CLK_MUX_FOR_20_MODE”, тогда, чтобы эта функция заработала, мультиплексирование (muxing -	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		уплотнение) синхросигналов/сигналов 2.0 в pipe_clk-синхросигналы/сигналы следует производить вне контроллера.		
[27]	DEV_TRB_OUT_SPR_IND	0: (поведение по умолчанию) dword статуса TRB остается без изменений. 1: функция подключена, в статусе OUT TRB указан короткий пакет. Этот бит применим только в режиме “Device” (и игнорируется в режиме “Host”). Если аппаратному/программному приложению устройства необходимо получать информацию о том, получен ли короткий пакет для передачи OUT, в самом статусе TRB, эта функция может быть подключена, и тогда бит будет установлен в отложенных записях TRB (writeback) в buf_size двойного слова dword. Bit[26] - SPR (сигнал о получении короткого пакета) двойного слова {trbstatus, RSVD, SPR, PCM1, bufsize} будет установлен в процессе передачи OUT в отложенных записях TRB, если это последний блок TRB, используемый для этого дескриптора передач. Этот бит квази-статичен, таким образом, он не должен изменяться в процессе работы устройства.	0x0	R/W
[28]	TX_IPGAP_LINECHECK_DIS	0: (работа по умолчанию) проверка состояния Linestate остаётся неизменной. 1: функция подключена: 2.0 MAC отключает проверку состояния Linestate в процессе высокоскоростной (HS) передачи. Этот бит применим для высокоскоростного (HS) режима работы u2mac. Если эта функция подключена, то 2.0 mac, работающий в высокоскоростном режиме, будет игнорировать состояние канала UTMI/ULPI Linestate в процессе передачи токена (во время IPGAP токен-к-токену и токен-к-данным). Когда это подключено, контроллер вставляет фиксированную 40-битную задержку TxEndDelay после того, как пакет выдан на UTMI и игнорирует Linestate в течение этого времени. Эта функция применима только в высокоскоростном (HS) режиме работы. Режим “Device”: если установлен GUCTL1.TX_IPGAP_LINECHECK_DIS, тогда для подтверждения согласования управления питанием каналов (LPM handshake) в устройстве система будет игнорировать linestate после TX и ожидать фиксированных синхросигналов (эквивалентных 40 тактам передачи) после передачи ACK на utmi. Режим “Host”: если установлен GUCTL1.TX_IPGAP_LINECHECK_DIS, тогда интервал ipgap между (tkn и tkn/data)	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		добавляется после 40 тактов передачи задержки TXENDDELAY, и на протяжении этой задержки в 40 тактов передачи linestate игнорируется. Следует подключать этот бит, если LineState не отражает ожидаемое состояние канала (J) в процессе передачи. Этот бит квази-статичен, таким образом, он не должен изменяться в процессе работы устройства.		
[29]	FILTER_SE0_FSLS_EOP	0: работа по умолчанию: проверка состояния канала Linestate для обнаружения SE0 в полноскоростном/низкоскоростном (FS/LS) режимах. 1: функция подключена: SE0 в полноскоростном/низкоскоростном (FS/LS) режимах фильтруется на протяжении двух тактов для обнаружения EOP. Этот бит применим в полноскоростном/низкоскоростном (FS/LS) режимах. Если эта функция подключена, тогда SE0 на канале linestate валидируется в процессе двух подряд идущих фронтов синхросигнала utmi/ulpi для обнаружения EOP. Эта функция применима только в полноскоростном (FS) состоянии в режиме “Device” и в полноскоростном/низкоскоростном (FS/LS) состояниях в режиме “Host”. Режим “Device”: полноскоростной (FS) - если GUCTL1.FILTER_SE0_FSLS_EOP установлен, тогда для согласования управления питанием каналов (LPM handshake), система будет игнорировать однократный сбой SE0 на linestate в процессе передачи. Только 2 или более сбоев SE0 будут считаться валидным EOP (end of packet) на полноскоростном порте. Режим “Host”: полноскоростной/низкоскоростной (FS/LS) - если GUCTL1.FILTER_SE0_FSLS_EOP установлен, тогда система будет игнорировать однократный сбой SE0 на linestate в процессе передачи. Только 2 или более сбоев SE0 будут считаться валидным EOP на полноскоростном/низкоскоростном порте. Следует подключать эту функцию, если на LineState встречаются сбои SE0 в процессе передачи. Этот бит квази-статичен, он не должен изменяться в процессе работы устройства.	0x0	R/W
[30]	DS_RXDET_MAX_TOUT_CTRL	Этот бит используется для управления таймером tRxDetectTimeoutDFP для сверхскоростного канала (SuperSpeed link).	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		0: (работа по умолчанию) таймаут tRxDetectTimeoutDFP имеет длительность 12 мс. 1: таймаут tRxDetectTimeoutDFP имеет длительность 120 мс. Этот бит используется только в режиме “Host”. Более подробное описание представлено в ECN020 в спецификации USB 3.0.		
[31]	DEV_DECOUPLE_L1L2_EVT	0: (работа по умолчанию) события устройства для L1/L2 U3 неизменны, события не разделяются (поведение, которое было реализовано в версиях v2.90a и ранее). 1: функция подключена, события L1 и L2 разделены при работе в режиме 2.0. Раздельные события позволяют битам для L1 приостанавливать и пробуждать события. Этот бит применим только в режиме “Device”. Если эта функция подключена, функции приостановки и пробуждения событий в L1 обладают индивидуальными настройками для подключения/маскировки событий. Рекомендуется подключать эту функцию если необходимо получать события L1 (LPM) раздельно, необъединенными с событиями L2 при работе на скоростях режима 2.0.	0	R/W

26.3.9 Глобальный регистр идентификатора Synopsys (GSNPSID)

Это регистр, доступный только для чтения, который содержит номер версии системы.

Поля регистра GSNPSID описаны в Таблица 26.19.

Таблица 26.19. Глобальный регистр идентификатора Synopsys

Диапазон	Название	Описание	Исходное значение	Тип доступа
[31:0]	SYNOPSYSID	Идентификатор Synopsys. SynopsysID[31:16] указывает номер идентификации системы (Core Identification Number). 0x5533 в ASCII для U3 (DWC_usb3). SynopsysID[15:0] указывает номер версии. Текущая версия 2.90a. Программное обеспечение использует этот регистр для конфигурирования в драйвере функций, специфичных для конкретных версий. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x5533320a	R

26.3.10 Глобальный регистр ввода/вывода общего назначения (GGPIO)

Приложение может использовать этот регистр для портов ввода и вывода общего назначения, или для отладки.

Поля регистра GGPIO описаны в Таблица 26.20.

Таблица 26.20. Глобальный регистр ввода/вывода общего назначения

Диапазон	Название	Описание	Исходное значение	Тип доступа
[15:0]	GPI	Ввод общего назначения. Значение этого поля при чтении отражает значение системного ввода gp_in[15:0]. Замечание: в процессе тестирования bit-bash на этом регистре не следует проверять значение сброса данного поля, поскольку оно непредсказуемо; зависит от порта gp_in. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R
[31:16]	GPO	Вывод общего назначения. Значение этого поля отображается на системном порте вывода gp_out[15:0].	0x0	R/W

26.3.11 Глобальный регистр идентификатора пользователя (GUID)

Регистр, содержащий идентификатор пользователя (User ID). Он доступен как на чтение, так и на запись.

Значение этого регистра при включении питания задано как «регистр идентификации пользователя». Значение при включении питания в процессе конфигурирования через coreConsultant: параметр DWC_USB3_USERID.

Этот регистр может использоваться следующим образом:

- для хранения номера версии или ревизии системы;
- для хранения аппаратных конфигураций, внешних по отношению к системе;
- как рабочий (черновой) регистр.

Поля регистра GUID описаны в Таблица 26.21.

Таблица 26.21. Глобальный регистр идентификатора пользователя

Диапазон	Название	Описание	Исходное значение	Тип доступа
[31:0]	USERID	Поле идентификатора, доступное для программирования приложениями.	0x12345678	R/W

26.3.12 Глобальный регистр пользовательских настроек (GUCTL)

Этот регистр предоставляет несколько способов управлять поведением системы с помощью программного обеспечения в режиме “Host”. Большая часть этих способов применяется для улучшения совместимости хоста с различными устройствами.

Поля регистра GUCTL описаны в Таблица 26.22.

Таблица 26.22. Глобальный регистр пользовательских настроек

Диапазон	Название	Описание	Исходное значение	Тип доступа
[8:0]	DTFT	Точная настройка таймаута устройства (DTFT). Это поле - параметр режима “Host”, который определяет время, в течение которого хост ожидает ответа от устройства до рассмотрения возможности таймаута. Для работы поля DTFT необходимо, чтобы в поле DTCT было установлено 2'b00. Значение поля DTFT - это 125 МГц * 256 - число тактов на частоте 125 МГц*256, которое следует отсчитать перед рассмотрением возможности таймаута прибора. Минимальное значение DTFT - 2. Например, если синхросигнал mac3_clk имеет частоту 125 МГц (период 8 нс), это поле вычисляется следующим образом: (значение DTFT) * 256 * (8 нс). Краткая справка: Если DTFT = 0x2, 2*256*8 = 4мкс таймаута. Если DTFT = 0x5, 5*256*8 = 10мкс таймаута. Если DTFT = 0xA, 10*256*8 = 20мкс таймаута. Если DTFT = 0x10, 16*256*8 = 32мкс таймаута. Если DTFT = 0x19, 25*256*8 = 51мкс таймаута. Если DTFT = 0x31, 49*256*8 = 100мкс таймаута. Если DTFT = 0x62, 98*256*8 = 200мкс таймаута. Замечания: - когда подключена сверхскоростная микросхема SuperSpeed Inter-Chip (SSIC), в режиме HS_G1_G2_G3, когда ssic_soc_pa_clk_freq = 2'b01, тогда значение DTFT следует вычислять для частоты 156.25 МГц, например, для эквивалентной задержки в 32 мкс значение DTFT должно быть 9'h14.	0x10	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		- когда системное запаздывание оказывается длиннее, чем запрограммированное значение DTCT/DTFT, то, если контроллер хоста не в состоянии принять определенные транзакции на шине (из-за задержек системной шины), контроллер может не передать информацию о заголовке (header credits), что, в свою очередь, может спровоцировать хост отправить отчёт об ошибке транзакции. Таким образом, это значение следует программировать большим, чем возможное запаздывание системы.		
[10:9]	DTCT	Грубая настройка таймаута устройства (DTCT). Это поле - параметр основного режима, который определяет время, в течение которого хост ожидает ответа от устройства до рассмотрения возможности таймаута. Система проверяет значение DTCT в первую очередь. Если оно равно «0», тогда значение таймаута определяется полем DTFT. Если оно ненулевое, то система использует следующие значения таймаута: - 2'b00: 0 мкс -> следует воспользоваться значением DTFT; - 2'b01: 500 мкс; - 2'b10: 1.5 мс; - 2'b11: 6.5 мс. Замечание: когда системное запаздывание оказывается длиннее, чем запрограммированное значение DTCT/DTFT, то, если контроллер хоста не в состоянии принять определенные транзакции на шине (из-за задержек системной шины), контроллер может не передать информацию о заголовке (header credits), что, в свою очередь, может спровоцировать хост отправить отчёт об ошибке транзакции. Таким образом, это значение следует программировать большим, чем возможное запаздывание системы.	0x0	R/W
[11]	InsrtExtrFSBODI	Вставка добавочной задержки между полноскоростными (FS) транзакциями Bulk OUT (InsrtExtrFSBODl). Некоторые полноскоростные устройства слишком медленны для получения данных типа Bulk OUT и могут зависнуть, когда происходят последовательные транзакции Bulk OUT с короткими интервалами задержек между ними. Этот бит используется, чтобы контролировать, должен ли хост	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		вставлять добавочные задержки между последовательными транзакциями Bulk OUT к конечной точке полноскоростного устройства (FS Endpoint). - 1'b0: хост не вставляет добавочную задержку между идущими подряд транзакциями Bulk OUT к конечной точке полноскоростного устройства. - 1'b1: хост вставляет добавочную задержку длиной около 12 мкс между идущими подряд транзакциями Bulk OUT к конечной точке полноскоростного устройства, чтобы обойти проблемы со скоростями устройств. Замечание: установка значения «1» в этот бит ухудшит производительность передачи Bulk OUT для большинства полноскоростных приборов.		
[12]	ExtCapSupptEN	Подключение поддержки внешних расширенных возможностей (ExtCapSuptEN). Когда установлен, этот бит разрешает реализацию расширенных возможностей вне системы. Когда ExtCapSupEN установлен и подключена возможность отладки (Debug Capability), указатель следующей возможности (Next Capability) в «возможности отладки» возвращает 16. Команда чтения первого DWORD последней внутренней расширенной возможности (в случае, если возможность отладки не подключена, команда чтения вернет “xHCI Supported Protocol Capability для USB 3.0”) возвращает значение 4 в поле указателя следующей возможности. Для программного обеспечения это является указанием на то, что существуют другие 4 слова DWORD для возможности после этой возможности (например, по адресу N+16, где N - адрес этого DWORD). Если это подключено, должен быть реализован внешний декодировщик адресов, который отслеживает подчиненный интерфейс xHCI. Если им замечены обращения к адресам N+16 или более, доступ подчиненного интерфейса перенаправляется к той части аппаратуры, которая возвращает регистр указателя внешней возможности	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		и также обеспечивает поддержку чтений/записей в эту новую возможность и все сопуствтующие ей эффекты. Если это отключено, команда чтения первого DWORD последней внутренней расширенной возможности возвращает 0 в поле указателя следующей возможности. Это указывает, что возможностей больше нет.		
[13]	EnOverlapChk	Подключает проверку наложения низкочастотного периодического сигнала (LFPS Overlap) во время удаленного выхода из режима Ux. Возможные значения: -1'b1: при выходе из режима U1/U2/U3, сверхскоростной (SuperSpeed) канал ожидает получения либо удаленного канала LFPS, либо тренинговых символов наборов TS1/TS2 перед тем, как подтвердить, что установка связи LFPS завершена. Это делается в целях предупреждения случая, в котором сбой сигнала LFPS заставит канал начать выход из состояния низкого питания. Поиск наложения сигнала LFPS позволяет удостовериться, что партнер канала тоже получает сигнал LFPS. -1'b0: когда канал выходит из режима U1/U2/U3 по причине удаленного выхода, он не производит поиск наложения сигнала LFPS.	0x0	R/W
[14]	USBHstInAutoRetryEn	Авто-повтор передач IN хоста (USBHstInAutoRetryEn). Когда установлено, это поле подключает функцию авто-повтора (Auto Retry). Для передач к хосту (IN transfers) (не-изохронных), которые сталкиваются с пакетами данных, содержащими ошибки CRC или сценарии внутренних перегрузок (internal overrun), функция автоповтора заставляет систему хоста послать прибору не прерывающий повторение пакет ACK (acknowledged) (таким образом, это будет пакет транзакции ACK со значениями Retry = 1 и NumP != 0). Если функция авто-повтора отключена (как в исходном состоянии), в этом случае система ответит прерывающим повторение пакетом ACK (таким образом, это будет пакет транзакции ACK со значениями Retry = 1 и NumP = 0). Возможные значения: - 1'b0: авто-повтор отключен; - 1'b1: авто-повтор подключен.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Замечание: при подключении функции авто-повтора, если системное запаздывание достаточно велико, чтобы спровоцировать заполнение очереди состояний протокола (Protocol Status Queue) (PSQ может быть заполнена в результате накопления необработанных сообщений из-за ожидающих обработки вызовов перед очисткой очереди TxQ в соответствии с условиями протоколов NRDY/ERDY (Retry, Not Ready/Endpoint Ready)), тогда контроллер хоста может сгенерировать ошибку передачи.		
[15]	reserved_15	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0x1.	0x0	R
[16]	ResBwHSEPS	Резервирует 85% пропускной способности (Bandwidth) для высокоскоростных (HS) периодических конечных точек (Periodic EPs)(ResBwHSEPS). По умолчанию, контроллер хоста резервирует 80% пропускной способности на периодические конечные точки. Если этот бит установлен, пропускная способность расширяется до 85%, чтобы вместить две высокоскоростных широкодиапазонных изохронных концевых точки (ISOC EPs). Стандарт USB 2.0 требовал отводить 80% пропускной способности под изохронный трафик. Если два широкодиапазонных изохронных прибора (веб-камеры с высоким разрешением, HD Webcams) соединены, и если каждый требует 1024 байта X 3 пакета на микрофрейм, тогда требуется ширина диапазона около 82%. Если этот бит установлен, тогда есть возможность соединить две веб-камеры, каждую с полезной нагрузкой 1024 байта X 3 на микрофрейм. В противном случае, может понадобиться уменьшить разрешение веб-камер. Этот бит валиден в конфигурации хоста и устройства двойного назначения (Host and DRD), и используется только при работе в режиме “Host”. В режиме “Device” этот бит следует игнорировать.	0x0	R/W
[17]	SprsCtrlTransEn	Подключение разреженных контрольных транзакций (Sparse Control Transaction Enable). Некоторые устройства отзываются на контрольные передачи достаточно медленно. Планирование множества транзакций в один фрейм/микрофрейм передачи может заставить эти	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		устройства функционировать неправильно. Если в этот бит установлено значение 1'b1, контроллер хоста планирует транзакции контрольной передачи в разных фреймах/микрофреймах.		
[20:18]	reserved_20_18	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0x7. Ограничение записи: writeAsRead.	0x0	R
[21]	NoExtrDl	Отсутствие добавочной задержки между пакетом начала фрейма (SOF) и первым пакетом (NoExtrDl). Некоторые высокоскоростные устройства функционируют неправильно, когда хост отправляет пакет сразу после пакета SOF. Однако, вставка добавочной задержки между пакетом SOF и первым пакетом может снизить пропускную способность и производительность шины USB. Этот бит управляет решением о том, должен ли хост ожидать 2 мкс перед отправкой первого пакета после отправки пакета SOF или нет. Пользователь может установить «1» в этот бит для улучшения работы, если в его среде окружения хоста нет устройств, вызывающих подобные затруднения. - 1'b0: хост ожидает 2 мкс после отправки пакета SOF перед тем, как отправить первый пакет USB; - 1'b1: хост не делает пауз после отправки пакета SOF перед тем, как отправить первый пакет USB.	0x0	R/W
[31:22]	REFCLKPER	Это поле указывает период синхросигнала ref_clk в наносекундах. Значение этого регистра по умолчанию установлено как 'h8 (8нс/125 МГц). Необходимо обновление этого поля в процессе инициализации при запуске, если в параметры GCTL.SOFITPSYNC или GFLADJ.GFLADJ_REFCLK_LPM_SEL установлена «1». Максимальное программируемое значение - 62 нс, а минимальное значение - 8 нс. Следует использовать опорный синхросигнал с периодом, который является целым кратным, так, чтобы пакет изохронной временной метки (ITP) укладывался в границу флуктуаций в 32 нс. Разрешаемые частоты ref_clk, период которых не является целым кратным - 16/17/19.2/24/39.7МГц.	0x8	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Это поле ни при каких условиях не должно получать значение «0». Если использование этой функции не планируется, следует установить в это поле значение по умолчанию, 'h8.		

26.3.13 Младший глобальный регистр ошибок адреса шины SoC (GBUSERRADDRLO)

Это альтернативный регистр для регистра GBUSERRADDR.

Поля регистра GBUSERRADDRLO описаны в Таблица 26.23.

Таблица 26.23. Младший глобальный регистр ошибок адреса шины SoC

Диапазон	Название	Описание	Исходное значение	Тип доступа
[31:0]	BUSERRADDR	Адрес шины - младшие биты (Bus Address - Low (BusAddrLo)). Этот регистр содержит младшие 32 бита первого адреса шины, в котором встречается ошибка шины SoC. Он валиден, когда поле GSTS.BusErrAddrVld имеет значение 1. Он может быть очищен только сбросом системы. Замечание: поддерживается только в конфигурациях AHB и AXI. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R

26.3.14 Старший глобальный регистр ошибок адреса шины SoC (GBUSERRADDRHI)

Это альтернативный регистр для регистра GBUSERRADDR.

Поля регистра GBUSERRADDRHI описаны в Таблица 26.24.

Таблица 26.24. Старший глобальный регистр ошибок адреса шины SoC

Диапазон	Название	Описание	Исходное значение	Тип доступа
[31:0]	BUSERRADDR	Адрес шины - старшие биты (Bus Address - High (BusAddrHi)). Этот регистр содержит старшие 32 бита первого адреса шины, в котором встречается ошибка шины SoC. Он валиден, когда поле GSTS.BusErrAddrVld имеет значение 1. Он может быть очищен только сбросом системы. Замечание: поддерживается только в конфигурациях AHB и AXI. Контролепригодность: writeAsRead.	0x0	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Ограничение записи: writeAsRead.		

26.3.15 Младший глобальный регистр установки соответствий сверхскоростного порта к экземпляру шины (GPRTBIMAPLO)

Это альтернативный регистр для регистра GPRTBIMAP.

Поля регистра имеют тип доступа read-write с учётом конструируемого номера порта. Ограничение writeAsRead добавлено для того, чтобы ограничить побочные эффекты для неиспользуемых полей.

Для конфигурации, в которой число портов USB 3.0 совпадает с числом экземпляров сверхскоростной шины (SS Bus Instances), не следует переопределять соответствия в режиме отладки. Если соответствия приходится переопределить, то хост отладки должен быть соединен с тем портом, который обладает назначенным ему экземпляром сверхскоростной шины.

Например, если DWC_USB3_NUM_U3_ROOT_PORTS =3 и DWC_USB3_NUM_SS_USB_INSTANCES=3, и программное обеспечение ставит первый сверхскоростной порт (SS port) в соответствие первому экземпляру сверхскоростной шины (SS BI), а второй/третий порт – второму экземпляру сверхскоростной шины, тогда хост отладки может быть подсоединен только к первому порту.

Замечание: значения сброса совпадают с соответствующими им в регистре GPRTBIMAP.

Поля регистра GPRTBIMAPLO описаны в Таблица 26.25.

Таблица 26.25. Младший глобальный регистр установки соответствий сверхскоростного порта к экземпляру шины

Диапазон	Название	Описание	Исходное значение	Тип доступа
[3:0]	BINUM1	BINUM1: Номер SS USB экземпляра для порта 1. Поле идентификатора, доступное для программирования приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W
[7:4]	BINUM2	BINUM2: Номер SS USB экземпляра для порта 2. Поле идентификатора, доступное для программирования приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W
[11:8]	BINUM3	BINUM3: Номер SS USB экземпляра для порта 3. Поле идентификатора, доступное для программирования приложением. Контролепригодность: writeAsRead.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Ограничение записи: writeAsRead.		
[15:12]	BINUM4	BINUM4: Номер SS USB экземпляра для порта 4. Поле идентификатора, доступное для программирования приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W
[19:16]	BINUM5	BINUM5: Номер SS USB экземпляра для порта 5. Поле идентификатора, доступное для программирования приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W
[23:20]	BINUM6	BINUM6: Номер SS USB экземпляра для порта 6. Поле идентификатора, доступное для программирования приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W
[27:24]	BINUM7	BINUM7: Номер SS USB экземпляра для порта 7. Поле идентификатора, доступное для программирования приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W
[31:28]	BINUM8	BINUM8: Номер SS USB экземпляра для порта 8. Поле идентификатора, доступное для программирования приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W

26.3.16 Старший глобальный регистр установки соответствий сверхскоростного порта к экземпляру шины (GPRTBIMAPHI)

Это альтернативный регистр для регистра GPRTBIMAP.

Для конфигурации, в которой число портов USB 3.0 совпадает с числом экземпляров сверхскоростной шины (SS Bus Instances), не следует переопределять соответствия в режиме отладки. Если соответствия приходится переопределить, то хост отладки должен быть соединен с тем портом, который обладает назначенным ему экземпляром сверхскоростной шины.

Например, если DWC_USB3_NUM_U3_ROOT_PORTS =3 и DWC_USB3_NUM_SS_USB_INSTANCES=3, и программное обеспечение ставит первый сверхскоростной порт (SS port) в соответствие первому экземпляру сверхскоростной шины (SS BI), а второй/третий порт – второму экземпляру сверхскоростной шины, тогда хост отладки может быть подсоединен только к первому порту.

Замечание: значения сброса совпадают с соответствующими им в регистре GPRTBIMAP.

Поля регистра GPRTBIMAPHI описаны в Таблица 26.26.

Таблица 26.26. Старший глобальный регистр установки соответствий сверхскоростного порта к экземпляру шины

Диапазон	Название	Описание	Исходное значение	Тип доступа
[3:0]	BINUM9	BINUM9: Номер SS USB экземпляра для порта 9. Поле идентификатора, доступное для программирования приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W
[7:4]	BINUM10	BINUM10: Номер SS USB экземпляра для порта 10. Поле идентификатора, доступное для программирования приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W
[11:8]	BINUM11	BINUM11: Номер SS USB экземпляра для порта 11. Поле идентификатора, доступное для программирования приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W
[15:12]	BINUM12	BINUM12: Номер SS USB экземпляра для порта 12. Поле идентификатора, доступное для программирования приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W
[19:16]	BINUM13	BINUM13: Номер SS USB экземпляра для порта 13. Поле идентификатора, доступное для программирования приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W
[23:20]	BINUM14	BINUM14: Номер SS USB экземпляра для порта 14. Поле идентификатора, доступное для программирования приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W
[27:24]	BINUM15	BINUM15: Номер SS USB экземпляра для порта 15. Поле идентификатора, доступное для программирования приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W
[31:28]	reserved_31_28	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0xf. Ограничение записи: writeAsRead.	0x0	R

26.3.17 Глобальный регистр аппаратных параметров 0 (GHWPARAMS0)

Этот регистр содержит опции аппаратной конфигурации, которые могут быть подключены через GUI в coreConsultant.

Описание каждого параметра представлено в главе «Параметры» справочника. Эта информация также доступна в coreConsultant: для этого необходимо кликнуть правой кнопкой мыши по ярлыку параметра и выбрать «Что это» (“What's This”); также можно воспользоваться вкладкой «Помощь».

Замечание: некоторые глобальные аппаратные параметры недоступны для модификации через coreConsultant в настоящий момент. Эти настройки находятся в файле <workspace>/src/DWC_usb3_params.v. Пользователь не должен их изменять.

Поля регистра GHWPARAMS0 описаны в Таблица 26.27.

Таблица 26.27. Глобальный регистр аппаратных параметров 0

Диапазон	Название	Описание	Исходное значение	Тип доступа
[2:0]	ghwparams0_2_0	`DWC_USB3_MODE. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x2	R
[5:3]	ghwparams0_5_3	`DWC_USB3_MBUS_TYPE. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x1	R
[7:6]	ghwparams0_7_6	`DWC_USB3_SBUS_TYPE. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x1	R
[15:8]	ghwparams0_15_8	`DWC_USB3_MDWIDTH. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x40	R
[23:16]	ghwparams0_23_16	`DWC_USB3_SDWIDTH. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x20	R
[31:24]	ghwparams0_31_24	`DWC_USB3_AWIDTH. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x40	R

26.3.18 Глобальный регистр аппаратных параметров 1 (GHWPARAMS1)

Этот регистр содержит опции аппаратной конфигурации, которые могут быть подключены через GUI в coreConsultant.

Описание каждого параметра представлено в главе «Параметры» справочника. Эта информация также доступна в coreConsultant: для этого необходимо кликнуть правой кнопкой мыши по ярлыку параметра и выбрать «Что это» (“What's This”); также можно воспользоваться вкладкой «Помощь».

Замечание: некоторые глобальные аппаратные параметры недоступны для модификации через coreConsultant в настоящий момент. Эти настройки находятся в файле <workspace>/src/DWC_usb3_params.v. Пользователь не должен их изменять.

Поля регистра GHWPARAMS1 описаны в Таблица 26.28.

Таблица 26.28. Глобальный регистр аппаратных параметров 1

Диапазон	Название	Описание	Исходное значение	Тип доступа
[2:0]	ghwparams1_2_0	`DWC_USB3_IDWIDTH-1. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x3	R
[5:3]	ghwparams1_5_3	`DWC_USB3_BURSTWIDTH-1. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x7	R
[8:6]	ghwparams1_8_6	`DWC_USB3_DATAINFOWIDTH. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x4	R
[11:9]	ghwparams1_11_9	`DWC_USB3_REQINFOWIDTH. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x4	R
[14:12]	ghwparams1_14_12	`DWC_USB3_ASPACEWIDTH. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x4	R
[20:15]	ghwparams1_20_15	`DWC_USB3_DEVICE_NUM_INT.  Подробнее `DWC_USB3_DEVICE_NUM_INT описано в файле <workspace>/src/DWC_usb3_params.v. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x1	R
[22:21]	ghwparams1_22_21	`DWC_USB3_NUM_RAMS. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x3	R
[23]	ghwparams1_23	`DWC_USB3_SPRAM_TYP. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R
[25:24]	ghwparams1_25_24	`DWC_USB3_EN_PWROPT. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x1	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
[26]	ghwparams1_26	`DWC_USB3_MAC_PHY_CLKS_SYNC. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R
[27]	ghwparams1_27	`DWC_USB3_MAC_RAM_CLKS_SYNC. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R
[28]	ghwparams1_28	`DWC_USB3_RAM_BUS_CLKS_SYNC. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R
[29]	ghwparams1_29	Зарезервировано. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R
[30]	ghwparams1_30	`DWC_USB3_RM_OPT_FEATURES. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R
[31]	ghwparams1_31	`DWC_USB3_EN_DBC. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x1	R

26.3.19 Глобальный регистр аппаратных параметров 2 (GHWPARAMS2)

Этот регистр содержит опции аппаратной конфигурации, которые могут быть подключены через GUI в coreConsultant.

Описание каждого параметра представлено в главе «Параметры» справочника. Эта информация также доступна в coreConsultant: для этого необходимо кликнуть правой кнопкой мыши по ярлыку параметра и выбрать «Что это» (“What's This”); также можно воспользоваться вкладкой «Помощь».

Замечание: некоторые глобальные аппаратные параметры недоступны для модификации через coreConsultant в настоящий момент. Эти настройки находятся в файле <workspace>/src/DWC_usb3_params.v. Пользователь не должен их изменять.

Поля регистра GHWPARAMS2 описаны в Таблица 26.29.

Таблица 26.29. Глобальный регистр аппаратных параметров 2

Диапазон	Название	Описание	Исходное значение	Тип доступа
[31:0]	ghwparams2_31_0	`DWC_USB3_USERID. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x12345678	R

26.3.20 Глобальный регистр аппаратных параметров 3 (GHWPARAMS3)

Этот регистр содержит опции аппаратной конфигурации, которые могут быть подключены через GUI в coreConsultant.

Описание каждого параметра представлено в главе «Параметры» справочника. Эта информация также доступна в coreConsultant: для этого необходимо кликнуть правой кнопкой мыши по ярлыку параметра и выбрать «Что это» (“What's This”); также можно воспользоваться вкладкой «Помощь».

Замечание: некоторые глобальные аппаратные параметры недоступны для модификации через coreConsultant в настоящий момент. Эти настройки находятся в файле <workspace>/src/DWC_usb3_params.v. Пользователь не должен их изменять.

Поля регистра GHWPARAMS3 описаны в Таблица 26.30.

Таблица 26.30. Глобальный регистр аппаратных параметров 3

Диапазон	Название	Описание	Исходное значение	Тип доступа
[1:0]	ghwparams3_1_0	`DWC_USB3_SSPHY_INTERFACE. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x1	R
[3:2]	ghwparams3_3_2	`DWC_USB3_HSPHY_INTERFACE. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x1	R
[5:4]	ghwparams3_5_4	`DWC_USB3_FSPHY_INTERFACE. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R
[7:6]	ghwparams3_7_6	`DWC_USB3_HSPHY_DWIDTH. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x2	R
[9:8]	ghwparams3_9_8	Зарезервировано. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R
[10]	ghwparams3_10	`DWC_USB3_VENDOR_CTL_INTERFACE. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R
[11]	ghwparams3_11	`DWC_USB3_ULPI_CARKIT. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R
[17:12]	ghwparams3_17_12	`DWC_USB3_NUM_EPS. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x8	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
[22:18]	ghwparams3_22_18	`DWC_USB3_NUM_IN_EPS. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x4	R
[30:23]	ghwparams3_30_23	`DWC_USB3_CACHE_TOTAL_XFER_RESOURCES. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x8	R
[31]	ghwparams3_31	Зарезервировано. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R

26.3.21 Глобальный регистр аппаратных параметров 4 (GHWPARAMS4)

Этот регистр содержит опции аппаратной конфигурации, которые могут быть подключены через GUI в coreConsultant.

Описание каждого параметра представлено в главе «Параметры» справочника. Эта информация также доступна в coreConsultant: для этого необходимо кликнуть правой кнопкой мыши по ярлыку параметра и выбрать «Что это» (“What's This”); также можно воспользоваться вкладкой «Помощь».

Замечание: некоторые глобальные аппаратные параметры недоступны для модификации через coreConsultant в настоящий момент. Эти настройки находятся в файле <workspace>/src/DWC_usb3_params.v. Пользователь не должен их изменять.

Поля регистра GHWPARAMS4 описаны в Таблица 26.31.

Таблица 26.31. Глобальный регистр аппаратных параметров 4

Диапазон	Название	Описание	Исходное значение	Тип доступа
[5:0]	ghwparams4_5_0	`DWC_USB3_CACHE_TRBS_PER_TRANSFER. Контролепригодность: writeAsRead. Маска сброса: 0x3f. Ограничение записи: writeAsRead.	0x4	R
[6]	ghwparams4_6	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0x1. Ограничение записи: writeAsRead.	0x0	R
[8:7]	ghwparams4_8_7	`DWC_USB3_NUM_SSIC_NUM_LANE.  Этот бит указывает значение параметра `DWC_USB3_SSIC_NUM_LANE, выбранное пользователем: 2'b00: тракт 4;	0x0	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
		2'b01: тракт 1; 2'b10: тракт 2; 2'b11: зарезервировано. Замечание: когда в параметр DWC_USB3_NUM_SSIC_PORTS установлен 0, это поле является зарезервированным. Контролепригодность: writeAsRead. Маска сброса: 0x3. Ограничение записи: writeAsRead.		
[10:9]	ghwparams4_10_9	`DWC_USB3_SSIC_GEAR. Этот бит указывает значение параметра DWC_USB3_SSIC_GEAR, выбранное пользователем: 2'b00: зарезервировано; 2'b01: HS-G1; 2'b10: HS-G2; 2'b11: HS-G3. Замечание: когда в параметр DWC_USB3_NUM_SSIC_PORTS установлен 0, это поле является зарезервированным. Контролепригодность: writeAsRead. Маска сброса: 0x3. Ограничение записи: writeAsRead.	0x0	R
[11]	ghwparams4_11	`DWC_USB3_SSIC_NON_SNPS_MPHY. Это поле указывает, используется ли с портами SSIC M-PHY от Synopsys или M-PHY стороннего производителя. 1'b0: M-PHY от Synopsys; 1'b1: сторонняя M-PHY. Замечание: когда в параметр DWC_USB3_NUM_SSIC_PORTS установлен 0, это поле является зарезервированным. Контролепригодность: writeAsRead. Маска сброса: 0x1. Ограничение записи: writeAsRead.	0x0	R
[12]	ghwparams4_12	`DWC_USB3_EN_SSIC. 1'b0: если DWC_USB3_EN_SSIC == 0; 1'b1: если DWC_USB3_EN_SSIC != 0. Замечание: когда в параметр DWC_USB3_NUM_SSIC_PORTS установлен 0, это поле является зарезервированным. Контролепригодность: writeAsRead. Маска сброса: 0x1. Ограничение записи: writeAsRead.	0x0	R
[16:13]	ghwparams4_16_13	`DWC_USB3_HIBER_SCRATCHBUFS.	0x1	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Количество внешних буферов блокнотной памяти, которое запрашивает система для сохранения ее внутреннего состояния в режиме “Device”. Предполагается, что каждый буфер имеет объем 4 Кбайта. Массив буферов блокнотной памяти должен обладать этим количеством буферных указателей. Контролепригодность: writeAsRead. Маска сброса: 0xf. Ограничение записи: writeAsRead.		
[20:17]	ghwparams4_20_17	`DWC_USB3_NUM_SS_USB_INSTANCES. Контролепригодность: writeAsRead. Маска сброса: 0xf. Ограничение записи: writeAsRead.	0x1	R
[21]	ghwparams4_21	`DWC_USB3_EXT_BUFF_CONTROL. Контролепригодность: writeAsRead. Маска сброса: 0x1. Ограничение записи: writeAsRead.	0x0	R
[22]	ghwparams4_22	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0x1. Ограничение записи: writeAsRead.	0x0	R
[23]	ghwparams4_23	`DWC_USB3_EN_ISOC_SUPT. Контролепригодность: writeAsRead. Маска сброса: 0x1. Ограничение записи: writeAsRead.	0x1	R
[27:24]	ghwparams4_27_24	`DWC_USB3_BMU_PTL_DEPTH-1. Контролепригодность: writeAsRead. Маска сброса: 0xf. Ограничение записи: writeAsRead.	0x7	R
[31:28]	ghwparams4_31_28	`DWC_USB3_BMU_LSP_DEPTH. Контролепригодность: writeAsRead. Маска сброса: 0xf. Ограничение записи: writeAsRead.	0x4	R

26.3.22 Глобальный регистр аппаратных параметров 5 (GHWPARAMS5)

Этот регистр содержит опции аппаратной конфигурации, которые могут быть подключены через GUI в coreConsultant.

Описание каждого параметра представлено в главе «Параметры» справочника. Эта информация также доступна в coreConsultant: для этого необходимо кликнуть правой кнопкой мыши по ярлыку параметра и выбрать «Что это» (“What's This”); также можно воспользоваться вкладкой «Помощь».

Замечание: некоторые глобальные аппаратные параметры недоступны для модификации через coreConsultant в настоящий момент. Эти настройки находятся в файле <workspace>/src/DWC_usb3_params.v. Пользователь не должен их изменять.

Поля регистра GHWPARAMS5 описаны в Таблица 26.32.

Таблица 26.32. Глобальный регистр аппаратных параметров 5

Диапазон	Название	Описание	Исходное значение	Тип доступа
[3:0]	ghwparams5_3_0	`DWC_USB3_BMU_BUSGM_DEPTH. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x8	R
[9:4]	ghwparams5_9_4	`DWC_USB3_RXQ_FIFO_DEPTH. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x10	R
[15:10]	ghwparams5_15_10	`DWC_USB3_TXQ_FIFO_DEPTH. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x10	R
[21:16]	ghwparams5_21_16	`DWC_USB3_DWQ_FIFO_DEPTH. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x20	R
[27:22]	ghwparams5_27_22	`DWC_USB3_DFQ_FIFO_DEPTH. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x10	R
[31:28]	ghwparams5_31_28	Зарезервировано. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R

26.3.23 Глобальный регистр аппаратных параметров 6 (GHWPARAMS6)

Этот регистр содержит опции аппаратной конфигурации, которые могут быть подключены через GUI в coreConsultant.

Описание каждого параметра представлено в главе «Параметры» справочника. Эта информация также доступна в coreConsultant: для этого необходимо кликнуть правой кнопкой мыши по ярлыку параметра и выбрать «Что это» (“What's This”); также можно воспользоваться вкладкой «Помощь».

Замечание: некоторые глобальные аппаратные параметры недоступны для модификации через coreConsultant в настоящий момент. Эти настройки находятся в файле <workspace>/src/DWC_usb3_params.v. Пользователь не должен их изменять.

Поля регистра GHWPARAMS6 описаны в Таблица 26.33.

Таблица 26.33. Глобальный регистр аппаратных параметров 6

Диапазон	Название	Описание	Исходное значение	Тип доступа
[5:0]	ghwparams6_5_0	`DWC_USB3_PSQ_FIFO_DEPTH. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x20	R
[6]	ghwparams6_6	`DWC_USB3_EN_DBG_PORTS. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R
[7]	ghwparams6_7	`DWC_USB3_EN_FPGA. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R
[9:8]	ghwparams6_9_8	Зарезервировано. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R
[10]	SRPSupport	Подключена поддержка SRP (Session Request Protocol – протокол запроса сеанса). Приложение использует этот бит для того, чтобы определить, поддерживает ли система DWC_usb3 протокол SRP. 1'b0: поддержка SRP не подключена; 1'b1: поддержка SRP подключена; Этот бит имеет значение 1'b1, когда параметр DWC_USB3_EN_OTG имеет ненулевое значение. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R
[11]	HNPSupport	Подключена поддержка RSP/HNP (Role Swap Protocol/Host Negotiation Protocol - протоколы обмена ролями и согласования роли хоста). Приложение использует этот бит, чтобы определить, поддерживает ли система DWC_usb3 протоколы RSP/HNP. Если DWC_USB3_EN_OTG=2: 1'b0: поддержка RSP и HNP не подключена. Единственное исключение из этого правила существует для устройств SSPC-OTG, где поддержка RSP не подключена, но поддержка HNP подключена. (определить это можно, обратившись к биту OCFG.SSPC-OTG); 1'b1: поддержка RSP и HNP подключена. Если DWC_USB3_EN_OTG=1: 1'b0: поддержка HNP не подключена; 1'b1: поддержка HNP подключена. Этот бит подключен только если в coreConsultant режим HNP задан для режима работы (Mode of Operation) HNP (то есть,	0x0	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
		параметр DWC_USB3_EN_OTG имеет ненулевое значение, и DWC_USB3_MODE является устройством, исполняющим две роли (DRD - Dual Role Device)). В противном случае, он читается как 0. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.		
[12]	ADPSupport	`DWC_USB3_EN_ADP. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R
[13]	OTG_SS_Support	Подключена поддержка OTG 3.0. Этот бит указывает, установлено ли в параметр `DWC_USB3_EN_OTG значение 2. Другими словами, он указывает, поддерживает ли система DWC_usb3 OTG 3.0. 1'b0: поддержки OTG 3.0 нет; 1'b1: поддержка OTG 3.0 есть. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R
[14]	BCSupport	`DWC_USB3_EN_BC. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R
[15]	BusFltrsSupport	`DWC_USB3_EN_BUS_FILTERS. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x1	R
[31:16]	ghwparams6_31_16	`DWC_USB3_RAM0_DEPTH. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x88a	R

26.3.24 Глобальный регистр аппаратных параметров 7 (GHWPARAMS7)

Этот регистр содержит опции аппаратной конфигурации, которые могут быть подключены через GUI в coreConsultant.

Описание каждого параметра представлено в главе «Параметры» справочника. Эта информация также доступна в coreConsultant: для этого необходимо кликнуть правой кнопкой мыши по ярлыку параметра и выбрать «Что это» (“What's This”); также можно воспользоваться вкладкой «Помощь».

Замечание: некоторые глобальные аппаратные параметры недоступны для модификации через coreConsultant в настоящий момент. Эти настройки находятся в файле <workspace>/src/DWC_usb3_params.v. Пользователь не должен их изменять.

Поля регистра GHWPARAMS7 описаны в Таблица 26.34.

Таблица 26.34. Глобальный регистр аппаратных параметров 7

Диапазон	Название	Описание	Исходное значение	Тип доступа
[15:0]	ghwparams7_15_0	`DWC_USB3_RAM1_DEPTH. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x44d	R
[31:16]	ghwparams7_31_16	`DWC_USB3_RAM2_DEPTH. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x308	R

26.3.25 Глобальный регистр доступности пространства буфера FIFO/очереди отладки (GDBGFIFOSPACE)

Тест Bit Bash не должен применяться на этом регистре отладки.

Поля регистра GDBGFIFOSPACE описаны в Таблица 26.35.

Таблица 26.35. Глобальный регистр доступности пространства буфера FIFO/очереди отладки

Диапазон	Название	Описание	Исходное значение	Тип доступа
[8:0]	FIFO_QUEUE_SELECT	Выбор очереди/буфера FIFO или выбор порта (Port-Select). - Выбор очереди/буфера FIFO[8:5] указывает тип очереди/FIFO; - Выбор очереди/буфера FIFO[4:0] указывает номер очереди/FIFO. Например, 9'b0_0010_0001 обращается к RxFIFO_1 и 9'b0_0101_1110 обращается к TxReqQ_30. - От 9'b0_0001_1111 до 9'b0_0000_0000: от TxFIFO_31 до TxFIFO_0; - от 9'b0_0011_1111 до 9'b0_0010_0000: от RxFIFO_31 до RxFIFO_0; - от 9'b0_0101_1111 до 9'b0_0100_0000: от TxReqQ_31 до TxReqQ_0; - от 9'b0_0111_1111 до 9'b0_0110_0000: от RxReqQ_31 до RxReqQ_0; - от 9'b0_1001_1111 до 9'b0_1000_0000: от RxInfoQ_31 до RxInfoQ_0; - 9'b0_1010_0000: DescFetchQ_0 (для обратной совместимости);	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		- 9'b0_1010_0001: EventQ_0 (для обратной совместимости); - 9'b0_1010_0010: ProtocolStatusQ_0; - от 9'b0_1101_1111 до 9'b0_1110_0000: от DescFetchQ_31 до DescFetchQ_0; - от 9'b0_1111_1111 до 9'b0_1110_0000: от WriteBack/EventQ_31 до WriteBack/EventQ_0; - от 9'b1_0000_0111 до 9'b1_0000_0000: от AuxEventQ_7 до AuxEventQ_0 (если EN_SEPARATE_DESC_QUEUES=1). Port-Select[3:0] - выбирает номер порта при обращении к регистру GDBGLTSSM. Контролепригодность: нетестируемо. Маска сброса: 0x1ff.		
[15:9]	reserved_15_9	Зарезервировано. Контролепригодность: нетестируемо. Маска сброса: 0x7f.	0x0	R
[31:16]	SPACE_AVAILABLE	Контролепригодность: нетестируемо. Маска сброса: 0xffff.	0x42	R

26.3.26 Глобальный регистр отладки LTSSM (GDBGLTSSM)

В мульти-портовой конфигурации хоста, номер порта определяется полем Port-Select[3:0] в регистре GDBGFIFOSPACE.

Замечание:

- в режиме, использующем только USB 2.0, регистр GDBGLTSSM неприменим;
- тест Bit Bash не следует применять на этом регистре отладки.

Поля регистра GDBGLTSSM описаны в Таблица 26.36.

Таблица 26.36. Глобальный регистр отладки LTSSM

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	TXONESZEROS	Единицы/нули RTx (RTx Ones/Zeros (TxOnesZeros)). Описание этого поля представлено в таблице 5-3 спецификации PIPE3.	0x0	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Контролепригодность: нетестируемо. Маска сброса: 0x1.		
[1]	RXTERMINATION	Терминация Rx (Rx Termination (RxTermination)). Подробнее регистр `DWC_USB3_PIPE_RXTERM_RESET_VAL описан в <workspace>/src/DWC_usb3_params.v. Контролепригодность: нетестируемо. Маска сброса: 0x1.	0x0	R
[2]	TXSWING	Амплитуда Tx (Tx Swing). Подробнее данное поле описано в таблице 5-3 спецификации PIPE3. Контролепригодность: нетестируемо. Маска сброса: 0x1	0x0	R
[5:3]	LTDBClkState	Состояние синхросигнала LTSSM (Link Training and Status State Machine - машина тестирования каналов и состояния статуса) (LTSSM Clock State (LTDBClkState)). В мульти-портовой конфигурации хоста, номер порта определяется полем Port-Select[3:0] в регистре GDBGFIFOSPACE. Замечание: регистр GDBGLTSSM не применим в режиме работы только с USB 2.0. 000: CLK_NORM (PHY находится в любом состоянии, кроме P3 и работает сигнал PCLK); 001: CLK_TO_P3 (запрос к PHY на вход в P3); 010: CLK_WAIT1 (ожидание Phy_Status (запрос P3)); 011: CLK_P3 (PHY находится в состоянии P3 и сигнал PCLK не работает); 100: CLK_TO_P0 (запрос выхода из P3 к PHY); 101: CLK_WAIT2 (ожидание Phy_Status (запрос выхода из P3)). Контролепригодность: нетестируемо. Маска сброса: 0x7.	0x0	R
[7:6]	TXDEEMPHASIS	TXDEEMPHASIS (TxDeemphasis). Подробнее данное поле описано в таблице 5-3 спецификации PIPE3. Контролепригодность: нетестируемо. Маска сброса: 0x2.	0x1	R
[8]	RXEQTRAIN	Тестирование RxEq (RxEq Train). Подробнее данное поле описано в таблице 5-3 спецификации PIPE3. Контролепригодность: нетестируемо. Маска сброса: 0x1.	0x0	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
[10:9]	POWERDOWN	Отключение питания (PowerDown). Подробнее данное поле описано в таблице 5-3 спецификации PIPE3. Контролепригодность: нетестируемо. Маска сброса: 0x1.	0x2	R
[13:11]	LTDBPhyCmdState	Состояния команды LTSSM PHY (LTSSM PHY command State (LTDBPhyCmdState)). 000: PHY_IDLE (состояние команды PHY находится в простое (IDLE). Нет ожидающих обработки запросов к PHY); 001: PHY_DET (запрос на запуск обнаружения приемника (Receiver detection)); 010: PHY_DET_3 (ожидание Phy_Status (обнаружение приемника)); 011: PHY_PWR_DLY (задержка Pipe3_PowerDown P0 -> запрос на P1/P2/P3); 100: PHY_PWR_A (задержка для внутренней логики); 101: PHY_PWR_B (ожидание для Phy_Status (запрос на изменение состояния питания (Power state))). Контролепригодность: нетестируемо. Маска сброса: 0x7.	0x0	R
[14]	TxDetRxLoopback	Tx обнаруживает Rx/Обратная петля (Tx Detect Rx/Loopback) Подробнее данное поле описано в таблице 5-3 спецификации PIPE3. Контролепригодность: нетестируемо. Маска сброса: 0x1.	0x0	R
[15]	RXPOLARITY	Полярность Rx (Rx Polarity). Подробнее данное поле описано в таблице 5-3 спецификации PIPE3. Контролепригодность: нетестируемо. Маска сброса: 0x1.	0x0	R
[16]	TXELECLDLE	Электрический простой Tx (Tx Elec Idle). Подробнее данное поле описано в таблице 5-3 спецификации PIPE3. Контролепригодность: нетестируемо. Маска сброса: 0x0.	0x1	R
[17]	ELASTICBUFFERMODE	Режим эластичного буфера (Elastic Buffer Mode). Подробнее данное поле описано в таблице 5-3 спецификации PIPE3. Контролепригодность: нетестируемо. Маска сброса: 0x1.	0x0	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
[21:18]	LTDBSUBSTATE	Суб-состояние LTDB (LTDB Sub-State). Контролепригодность: нетестируемо. Маска сброса: 0xf.	0x0	R
[25:22]	LTDBLINKSTATE	Состояние канала LTDB (LTDB Link State). Контролепригодность: нетестируемо. Маска сброса: 0xb.	0x4	R
[26]	LTDBTIMEOUT	Таймаут LTDB (LTDB Timeout). Контролепригодность: нетестируемо. Маска сброса: 0x1.	0x0	R
[27]	PRTDIRECTION	Направление порта (Port Direction). 1'b0: входной поток (Upstream); 1'b1: выходной поток (Downstream); Замечание: этот бит примерим только для режима работы OTG 3.0 и зарезервирован для остальных режимов. Контролепригодность: нетестируемо. Маска сброса: 0x1.	0x0	R
[28]	X3_DS_HOST_SHUTDOWN	Ra3_ds_host_shutdown/b3_ds_host_shutdown. Замечание: этот бит примерим только для режима работы OTG 3.0 и зарезервирован для остальных режимов. Контролепригодность: нетестируемо. Маска сброса: 0x1.	0x0	R
[29]	X3_XS_SWAPPING	Ra3_ds_swapping/a3_us_swapping/b3_ds_swapping/b3_us_swapping. Следует интерпретировать это поле на основе направления порта. Замечание: этот бит примерим только для режима работы OTG 3.0 и зарезервирован для остальных режимов. Контролепригодность: нетестируемо. Маска сброса: 0x1.	0x0	R
[30]	RxElecidle	Электрический простой Rx (Rx Elec idle). Описание поля RxElecIdle доступно в таблице 5-4, «Сигналы статуса интерфейса (Status Interface Signals)» спецификации PIPE3. Контролепригодность: нетестируемо. Маска сброса: 0x0.	0x1	R
[31]	reserved_31_31	Зарезервировано. Контролепригодность: нетестируемо. Маска сброса: 0x1.	0x0	R

26.3.27 Глобальный регистр отладки LNMCC (GDBGLNMCC)

Тест Bit Bash не должен применяться на этом регистре отладки.

Поля регистра GDBGLNMCC описаны в Таблица 26.37.

Таблица 26.37. Глобальный регистр отладки LNMCC

Диапазон	Название	Описание	Исходное значение	Тип доступа
[8:0]	LNMCC_BERC	Это поле указывает информацию по объему битовых ошибок для порта, заданного в поле GDBGFIFOSPACE.PortSelect. Это поле предназначено только для отладки. Контролепригодность: нетестируемо. Маска сброса: 0x1ff. Ограничение записи: writeAsRead.	0x0	R
[31:9]	reserved_31_9	Зарезервировано. Контролепригодность: нетестируемо. Маска сброса: 0x7fffff.	0x0	R

26.3.28 Глобальный регистр отладки BMU (Buffer Mangement Unit - модуль управления буферами) (GDBGBMU)

Тест Bit Bash не должен применяться на этом регистре отладки.

Поля регистра GDBGBMU описаны в Таблица 26.38.

Таблица 26.38. Глобальный регистр отладки BMU (Buffer Mangement Unit - модуль управления буферами)

Диапазон	Название	Описание	Исходное значение	Тип доступа
[3:0]	BMU_CCU	Информация по отладке BMU_CCU. Контролепригодность: нетестируемо. Маска сброса: 0xf.	0x0	R
[7:4]	BMU_DCU	Информация по отладке BMU_DCU. Контролепригодность: writeAsRead. Маска сброса: 0xf.	0x0	R
[31:8]	BMU_BCU	Информация по отладке BMU_BCU. Контролепригодность: нетестируемо. Маска сброса: 0xffffff.	0x0	R

26.3.29 Глобальный регистр отладки LSP MUX для устройства (GDBGLSPMUX_DEV)

Этот регистр предназначен только для внутреннего использования.

Если DWC_USB3_PRESERVE_LOGIC_ANALYZER_SELECT подключен в процессе конфигурирования системы, тогда вывод значений по умолчанию (default values readout) – X (неопределен).

Тест Bit Bash не должен применяться на этом регистре отладки.

Поля регистра GDBGLSPMUX_DEV описаны в Таблица 26.39.

Таблица 26.39. Глобальный регистр отладки LSP MUX для устройства

Диапазон	Название	Описание	Исходное значение	Тип доступа
[3:0]	EPSELECT	Выбор конечной точки устройства (Device Endpoint Select). Выбирает информацию по отладке конечной точки, представленную в регистрах GDBGEPINFO в режиме “Device”. Контролепригодность: нетестируемо. Маска сброса: 0xf.	0x0	R/W
[7:4]	DEVSELECT	Выбор LSP (List Processor) для устройства (Device LSP Select). Выбирает информацию по отладке LSP, представленную в регистре GDBGLSP для режима “Device”. Контролепригодность: нетестируемо. Маска сброса: 0xf.	0x0	R/W
[13:8]	HOSTSELECT	Выбор LSP для хоста (Host LSP Select). Контролепригодность: нетестируемо. Маска сброса: 0x3f.	0x0	R/W
[14]	reserved_14	Зарезервировано. Контролепригодность: нетестируемо. Маска сброса: 0x1.	0x0	R
[15]	EnDbc	Подключение отладки возможности отладки LSP в режиме “Host”. Следует использовать HostSelect для выбора информации по отладке возможности отладки (DbC, Debug Capability) LSP, представленной в регистре GDBGLSP. Контролепригодность: нетестируемо. Маска сброса: 0x1.	0x0	R/W
[23:16]	logic_analyzer_trace	Выбор мультиплексора (MUX) порта протоколирования логического анализатора (Logic Analyzer Trace Port MUX Select). В настоящий момент используются только биты [21:16]. Подробнее о том, как мультиплексор управляет протоколами отладки, описано в разделе кода “assign logic_analyzer_trace =” в файле DWC_usb3.v.	0x3f	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Значение 6'h3F - устанавливает 0 на сигнал logic_analyzer_trace. Если планируется произвести операцию логического ИЛИ (OR) (используя мультиплексор вместо этого) этого сигнала с другими сигналами протоколирования в системе, чтобы сгенерировать общий сигнал протоколирования, можно использовать эту функцию. Контролепригодность: нетестируемо. Маска сброса: 0xc0.		
[31:24]	reserved_31_24	Зарезервировано. Контролепригодность: нетестируемо. Маска сброса: 0xff.	0x0	R

26.3.30 Глобальный регистр отладки LSP MUX для хоста (GDBGLSPMUX_HST)

Этот регистр предназначен только для внутреннего использования.

Если DWC_USB3_PRESERVE_LOGIC_ANALYZER_SELECT подключен в процессе конфигурирования системы, тогда вывод значений по умолчанию - X (неопределен).

Тест Bit Bash не должен применяться на этом регистре отладки.

Поля регистра GDBGLSPMUX_HST описаны в Таблица 26.40.

Таблица 26.40. Глобальный регистр отладки LSP MUX для хоста

Диапазон	Название	Описание	Исходное значение	Тип доступа
[13:0]	HOSTSELECT	Выбор LSP для устройства (Device LSP Select). Выбирает информацию по отладке LSP, представленную в регистре GDBGLSP для режима “Host”. Контролепригодность: нетестируемо. Маска сброса: 0xf.	0x0	R/W
[15:14]	reserved_15_14	Зарезервировано. Контролепригодность: нетестируемо. Маска сброса: 0x3.	0x0	R
[23:16]	logic_analyzer_trace	Выбор мультиплексора порта (Port MUX Select). В настоящий момент используются только биты [21:16]. Подробнее о том, как мультиплексор управляет протоколами отладки, описано в разделе кода “assign logic_analyzer_trace =” в файле DWC_usb3.v.	0x3f	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Значение 6'h3F - устанавливает 0 на сигнал logic_analyzer_trace. Если планируется произвести операцию логического ИЛИ (OR) (используя мультиплексор вместо этого) этого сигнала с другими сигналами протоколирования в системе, чтобы сгенерировать общий сигнал протоколирования, можно использовать эту функцию. Контролепригодность: нетестируемо. Маска сброса: 0xc0.		
[31:24]	reserved_31_24	Зарезервировано. Контролепригодность: нетестируемо. Маска сброса: 0xff.	0x0	R

26.3.31 Глобальный регистр отладки LSP (GDBGLSP)

Этот регистр предназначен исключительно для целей внутренней отладки.

Этот регистр предназначен исключительно для внутреннего пользования.

Если в процессе конфигурирования системы подключен DWC_USB3_PRESERVE_LOGIC_ANALYZER_SELECT, то значения по умолчанию будут читаться как X (неопределенно).

Не следует проводить тестирование Bit Bash на этом регистре отладки.

Поля регистра GDBGLSP описаны в Таблица 26.41.

Таблица 26.41. Глобальный регистр отладки LSP

Диапазон	Название	Описание	Исходное значение	Тип доступа
[31:0]	LSPDEBUG	Информация по отладке LSP. Контролепригодность: нетестируемо. Маска сброса: 0xffffffff.	0	R

26.3.32 Глобальный регистр информации по отладке конечной точки 0 (GDBGEPINFO0)

Этот регистр предназначен исключительно для внутреннего пользования.

Если в процессе конфигурирования системы подключен DWC_USB3_PRESERVE_LOGIC_ANALYZER_SELECT, то значения по умолчанию будут читаться как X (неопределено).

Не следует проводить тестирование Bit Bash на этом регистре отладки.

Поля регистра GDBGEPINFO0 описаны в Таблица 26.42.

Таблица 26.42. Глобальный регистр информации по отладке конечной точки 0

Диапазон	Название	Описание	Исходное значение	Тип доступа
[31:0]	EPDEBUG	Информация по отладке конечной точки, биты [31:0]. Контролепригодность: нетестируемо. Маска сброса: 0xffffffff.	0	R

26.3.33 Глобальный регистр информации по отладке конечной точки 1 (GDBGEPINFO1)

Этот регистр предназначен исключительно для внутреннего пользования.

Если в процессе конфигурирования системы подключен DWC_USB3_PRESERVE_LOGIC_ANALYZER_SELECT, то значения по умолчанию будут читаться как X (неопределено).

Не следует проводить тестирование Bit Bash на этом регистре отладки.

Поля регистра GDBGEPINFO1 описаны в Таблица 26.43.

Таблица 26.43. Глобальный регистр информации по отладке конечной точки 1

Диапазон	Название	Описание	Исходное значение	Тип доступа
[31:0]	EPDEBUG	Информация по отладке конечной точки, биты [63:32]. Контролепригодность: нетестируемо. Маска сброса: 0xffffffff.	0x800000	R

26.3.34 Младший глобальный регистр установки соответствий высокоскоростного порта к экземпляру шины (GPRTBIMAP_HSLO)

Это альтернативный регистр для регистра GPRTBIMAP_HS.

Поля регистра имеют тип доступа read-write с учётом конструируемого номера порта. Ограничение writeAsRead добавлено для того, чтобы ограничить побочные эффекты для неиспользуемых полей.

Замечание: значения сброса совпадают с соответствующими им в регистре GPRTBIMAP_HS.

Поля регистра GPRTBIMAP_HSLO описаны в Таблица 26.44.

Таблица 26.44. Младший глобальный регистр установки соответствий высокоскоростного порта к экземпляру шины

Диапазон	Название	Описание	Исходное значение	Тип доступа
[3:0]	BINUM1	BINUM1: Номер экземпляра HS USB для порта 1. Поле идентификатора, программируемое приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W
[7:4]	BINUM2	BINUM2: Номер экземпляра HS USB для порта 2. Поле идентификатора, программируемое приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W
[11:8]	BINUM3	BINUM3: Номер экземпляра HS USB для порта 3. Поле идентификатора, программируемое приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W
[15:12]	BINUM4	BINUM4: Номер экземпляра HS USB для порта 4. Поле идентификатора, программируемое приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W
[19:16]	BINUM5	BINUM5: Номер экземпляра HS USB для порта 5. Поле идентификатора, программируемое приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W
[23:20]	BINUM6	BINUM6: Номер экземпляра HS USB для порта 6. Поле идентификатора, программируемое приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W
[27:24]	BINUM7	BINUM7: Номер экземпляра HS USB для порта 7. Поле идентификатора, программируемое приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W
[31:28]	BINUM8	BINUM8: Номер экземпляра HS USB для порта 8. Поле идентификатора, программируемое приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W

26.3.35 Старший глобальный регистр установки соответствий высокоскоростного порта к экземпляру шины (GPRTBIMAP_HSHI)

Это альтернативный регистр для регистра GPRTBIMAP_HS.

Поля регистра имеют тип доступа read-write с учётом конструируемого номера порта. Ограничение writeAsRead добавлено для того, чтобы ограничить побочные эффекты для неиспользуемых полей.

Замечание: значения сброса совпадают с соответствующими им в регистре GPRTBIMAP_HS.

Поля регистра GPRTBIMAP_HSHI описаны в Таблица 26.45.

Таблица 26.45. Старший глобальный регистр установки соответствий высокоскоростного порта к экземпляру шины

Диапазон	Название	Описание	Исходное значение	Тип доступа
[3:0]	BINUM9	BINUM9: Номер экземпляра HS USB для порта 9. Поле идентификатора, программируемое приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W
[7:4]	BINUM10	BINUM10: Номер экземпляра HS USB для порта 10. Поле идентификатора, программируемое приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W
[11:8]	BINUM11	BINUM11: Номер экземпляра HS USB для порта 11. Поле идентификатора, программируемое приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W
[15:12]	BINUM12	BINUM12: Номер экземпляра HS USB для порта 12. Поле идентификатора, программируемое приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W
[19:16]	BINUM13	BINUM13: Номер экземпляра HS USB для порта 13. Поле идентификатора, программируемое приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W
[23:20]	BINUM14	BINUM14: Номер экземпляра HS USB для порта 14. Поле идентификатора, программируемое приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
[27:24]	BINUM15	BINUM15: Номер экземпляра HS USB для порта 15. Поле идентификатора, программируемое приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W
[31:28]	reserved_31_28	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0xf. Ограничение записи: writeAsRead.	0x0	R

26.3.36 Младший глобальный регистр установки соответствий полноскоростного порта к экземпляру шины (GPRTBIMAP_FSLO)

Это альтернативный регистр для регистра GPRTBIMAP_FS.

Поля регистра имеют тип доступа read-write с учётом конструируемого номера порта. Ограничение writeAsRead добавлено для того, чтобы ограничить побочные эффекты для неиспользуемых полей.

Замечание: значения сброса совпадают с соответствующими им в регистре GPRTBIMAP_FS.

Поля регистра GPRTBIMAP_FSLO описаны в Таблица 26.46.

Таблица 26.46. Младший глобальный регистр установки соответствий полноскоростного порта к экземпляру шины

Диапазон	Название	Описание	Исходное значение	Тип доступа
[3:0]	BINUM1	BINUM1: Номер экземпляра FS USB для порта 1. Поле идентификатора, программируемое приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W
[7:4]	BINUM2	BINUM2: Номер экземпляра FS USB для порта 2. Поле идентификатора, программируемое приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W
[11:8]	BINUM3	BINUM3: Номер экземпляра FS USB для порта 3. Поле идентификатора, программируемое приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
[15:12]	BINUM4	BINUM4: Номер экземпляра FS USB для порта 4. Поле идентификатора, программируемое приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W
[19:16]	BINUM5	BINUM5: Номер экземпляра FS USB для порта 5. Поле идентификатора, программируемое приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W
[23:20]	BINUM6	BINUM6: Номер экземпляра FS USB для порта 6. Поле идентификатора, программируемое приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W
[27:24]	BINUM7	BINUM7: Номер экземпляра FS USB для порта 7. Поле идентификатора, программируемое приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W
[31:28]	BINUM8	BINUM8: Номер экземпляра FS USB для порта 8. Поле идентификатора, программируемое приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W

26.3.37 Старший глобальный регистр установки соответствий полноскоростного порта к экземпляру шины (GPRTBIMAP_FSHI)

Это альтернативный регистр для регистра GPRTBIMAP_FS.

Поля регистра имеют тип доступа read-write с учётом конструируемого номера порта. Ограничение writeAsRead добавлено для того, чтобы ограничить побочные эффекты для неиспользуемых полей.

Замечание: значения сброса совпадают с соответствующими им в регистре GPRTBIMAP_FS.

Поля регистра GPRTBIMAP_FSHI описаны в Таблица 26.47.

Таблица 26.47. Старший глобальный регистр установки соответствий полноскоростного порта к экземпляру шины

Диапазон	Название	Описание	Исходное значение	Тип доступа
[3:0]	BINUM9	BINUM9: Номер экземпляра FS USB для порта 9. Поле идентификатора, программируемое приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W
[7:4]	BINUM10	BINUM10: Номер экземпляра FS USB для порта 10. Поле идентификатора, программируемое приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W
[11:8]	BINUM11	BINUM11: Номер экземпляра FS USB для порта 11. Поле идентификатора, программируемое приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W
[15:12]	BINUM12	BINUM12: Номер экземпляра FS USB для порта 12. Поле идентификатора, программируемое приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W
[19:16]	BINUM13	BINUM13: Номер экземпляра FS USB для порта 13. Поле идентификатора, программируемое приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W
[23:20]	BINUM14	BINUM14: Номер экземпляра FS USB для порта 14. Поле идентификатора, программируемое приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W
[27:24]	BINUM15	BINUM15: Номер экземпляра FS USB для порта 15. Поле идентификатора, программируемое приложением. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W
[31:28]	reserved_31_28	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0xf. Ограничение записи: writeAsRead.	0x0	R

26.3.38 Глобальный регистр настроек пользователя 2 (GUCTL2)

Этот регистр предоставляет программному обеспечению несколько опций для управления поведением системы в режимах “Host” и “Device”. Большая часть этих возможностей используется для улучшения стыкуемости с разными хостами и устройствами.

Поля регистра GUCTL2 описаны в Таблица 26.48.

Таблица 26.48. Глобальный регистр настроек пользователя 2

Диапазон	Название	Описание	Исходное значение	Тип доступа
[4:0]	TxPingDuration	Максимальная длительность пинга передачи. Это поле относится к режиму “Device” и управляет максимальной длительностью периода, в который контроллер должен отправить PHY команду о передаче “Ping LFPS”. Программирование этого значения задает длительность “Ping LFPS”. Каждая единица значения считается как 8 нс. Например: значение «13» означает 104 нс.	0xd	R/W
[10:5]	RxPingDuration	Максимальная длительность пинга получения. Это поле относится к режиму “Host” и управляет максимальной длительностью периода, в который полученный сигнал LFPS должен быть обработан как “Ping LFPS”. Программирование этого значения задает максимальную длительность “Ping LFPS”. Каждая единица значения считается как 8 нс. Например: значение «32» означает 256 нс.	0x20	R/W
[11]	DisableCFC	Отключение функции возможности идентификатора соприкасающихся фреймов, реализованной в корректировочном списке xHCI (xHCI Errata). (Disable xHCI Errata Feature Contiguous Frame ID Capability) Это поле управляет функцией возможности идентификатора соприкасающихся фреймов, реализованной в корректировочном списке xHCI. Когда этот бит установлен, в 11-й бит xHCI HCCPARAMS1 будет установлено значение «0», что укажет, что CFC (Contiguous Frame ID Capability - возможность идентификатора соприкасающихся фреймов) не поддерживается. Эту функцию следует отключать только в том случае, если используемое приложение не может обработать события ошибок пропущенных служб (Missed Service Error) при изохронных передачах, и запаздывания системы достаточно велики, чтобы провоцировать ошибки пропущенных служб, даже если программное обеспечение работает в соответствии с правилами изохронных ограничений (Isochronous Thresholding rules).	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
[12]	EnableEpCacheEvict	Подключение очистки кэша конечных точек после управления потоком передачи данных (Flow Control) для конечных точек типа bulk (Enable Evicting Endpoint cache after Flow Control for bulk endpoints). В версии 3.00a было произведено улучшение производительности для удержания конечной точки типа bulk IN, неспособной к потоковой передаче (non-stream capable), в рамках кэша после контроля передачи данных хостом (flow control). Установка этого бита отключает это улучшение. Он должен использоваться только в целях отладки.	0x0	R/W
[13]	reserved_13	Зарезервировано для дальнейшего использования. Контролепригодность: writeAsRead. Маска сброса: 0x1. Ограничение записи: writeAsRead.	0x0	R
[14]	Rst_actbitlater	Подключает очистку бита активности команды для команды ENDXFER после завершения исполнения команды. Этот бит валиден только в режиме “Device”.	0x0	R/W
[18:15]	NOLOWPWRDUR	Длительность нахождения вне режима низкого питания (NOLOWPWRDUR). Это поле применимо только в режиме “Device” и игнорируется в режиме “Host”. После начала передачи на изохронную конечную точку ESS ISOC, приложение должно запрограммировать значения битов этого поля. Каждая единица бита отражает длительность в миллисекундах. Например, значение 3 означает 3 мс.	0x0	R/W
[25:19]	EN_HP_PM_TIMER	Это поле используется для того, чтобы задать новые таймеры HP и PM. Для подключения таймера PM следует установить «1» в бит GUCTL2[19]. Для подключения таймера HP следует установить «1» в бит GUCTL2[20]. Для того, чтобы задать точное значение таймера HP в микросекундах, следует использовать биты GUCTL2[25:21].	0x0	R/W
[31:26]	reserved_31_26	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0xff. Ограничение записи: writeAsRead.	0x0	R/W

26.3.39 Глобальный регистр конфигурирования PHY USB2 (GUSB2PHYCFG(#n) (for n = 0; n <= DWC_USB3_NUM_U2_ROOT_PORTS-1))

Перед началом любых транзакций как на шине SoC, так и на USB, приложение обязано запрограммировать значения этого регистра.

В конфигурациях, относящихся только к устройству, требуется только один регистр.

В режиме “Host” реализуется по регистру для каждого порта.

Смещение: 0xc200 + (i * 0x4), где i = 0..=[DWC_USB3_NUM_U2_ROOT_PORTS]-1.

Поля регистра GUSB2PHYCFG(#n) (for n = 0; n <= DWC_USB3_NUM_U2_ROOT_PORTS-1) описаны в Таблица 26.49.

Таблица 26.49. Глобальный регистр конфигурирования PHY USB2

Диапазон	Название	Описание	Исходное значение	Тип доступа
[2:0]	ToutCal	Калибровка таймаута в высокоскоростном/полноскоростном режимах (HS/FS Timeout Calibration). Количество синхросигналов PHY, указанное приложением в данном поле, умножается на фактор битового времени (bit-time factor). Этот фактор добавляется к длительности межпакетного таймаута в системе в высокоскоростном/полноскоростном режиме, что обусловлено добавочными задержками от PHY. Это может потребоваться, поскольку добавочные задержки от PHY при генерации условий состояния канала могут варьироваться для разных PHY. Стандартное значение таймаута USB для высокоскоростной (HS) работы - от 736 до 816 (включительно) тактов передачи бита (bit times). Стандартное значение таймаута USB для полноскоростной (FS) работы - от 16 до 18 (включительно) тактов передачи бита. Приложение обязано программировать это поле, основываясь на скорости соединения. Число тактов передачи бита, добавляемых на такт синхроимпульса PHY: для высокоскоростного (HS) режима: - один синхроимпульс PHY 30 МГц = 16 тактов передачи бита; - один синхроимпульс PHY 60 МГц = 8 тактов передачи бита; для полноскоростного (FS) режима: - один синхроимпульс PHY 30 МГц = 0.4 такта передачи бита; - один синхроимпульс PHY 60 МГц = 0.2 такта передачи бита;	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		- один синхроимпульс PHY 48 МГц = 0.25 такта передачи бита.		
[3]	PHYIF	Интерфейс PHY (PHY Interface). Если выбран интерфейс UTMI+, приложение использует этот бит, чтобы сконфигурировать систему для поддержки UTMI+ PHY с 8- или 16-битным интерфейсом. 1'b0: 8 бит; 1'b1: 16 бит. Для режима ULPI: 1'b0. Замечания: - все подключенные порты 2.0 должны иметь ту же частоту синхросигнала, что и частота синхросигнала нулевого порта Port0 (utmi_clk[0]). - 8-битный и 16-битный режимы UTMI не могут быть использованы вместе для разных портов в одно и то же время (таким образом, все порты должны одновременно находиться в 8-битном режиме, или все должны находиться в 16-битном). - если любой из портов USB 2.0 выбран для работы в качестве порта ULPI, тогда все порты USB 2.0 должны работать на частоте 60 МГц.	0x0	R/W
[4]	ULPI_UTMI_Sel	Выбор интерфейса ULPI или UTMI+ (ULPI_UTMI_Sel). Приложение использует этот бит для выбора интерфейса UTMI+ или ULPI. 1'b0: интерфейс UTMI+; 1'b1: интерфейс ULPI. Этот бит доступен для записи, только если интерфейсы UTMI+ and ULPI заданы как высокоскоростные интерфейсы PHY (High-Speed PHY Interface(s)) в конфигурации coreConsultant (DWC_USB3_HSPHY_INTERFACE = 3). В противном случае, этот бит доступен только для чтения и его значение зависит от интерфейса, выбранного через DWC_USB3_HSPHY_INTERFACE. Ограничения записи: writeAsRead.	0x0	R
[5]	FSINTF	Выбор полноскоростного последовательного интерфейса (Full-Speed Serial Interface Select). Приложение использует этот бит для того, чтобы выбрать, должен ли последовательный полноскоростной интерфейс передатчика USB 1.1 быть однонаправленным или двунаправленным. 1'b0: 6-контактный однонаправленный последовательный полноскоростной интерфейс. В этот бит устанавливается «0», если к нему производится доступ на чтение. 1'b1: 3-контактный двунаправленный последовательный полноскоростной интерфейс. В этот бит устанавливается «0», если к нему производится доступ на чтение.	0x0	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Замечание: последовательный полноскоростной интерфейс USB 1.1 не поддерживается. Этот бит всегда читается как 1'b0. Ограничения записи: writeAsRead.		
[6]	SUSPENDUSB20	Приостановка высокоскоростной/полноскоростной/низкоскоростной PHY в USB 2.0 (Suspend USB2.0 HS/FS/LS PHY). Когда этот бит установлен, USB2.0 PHY входит в режим приостановки, если условия приостановки валидны. Для конфигураций DRD/OTG, рекомендуется, чтобы в процессе конфигурирования в coreConsultant в этот бит был установлен «0». Если в этот бит установлена «1», тогда приложение обязано произвести очистку этого бита после сброса при запуске (power-on reset). После окончания инициализации системы приложение должно установить в него «1». Для всех остальных конфигураций в этот бит возможна установка «1» в процессе конфигурирования системы. Замечания: - в режиме “Host”, при сбросе, в этот бит установлена «1». Программное обеспечение может отменить этот бит после сброса. - в режиме “Device”, перед отправкой любой команды конечным точкам устройства при работе на скоростях 2.0, следует отключить этот бит и переподключить его после завершения команды. Если, когда устройство находится в режиме L2, отправка команды производится без отключения этого бита, и если mac2_clk (utmi_clk/ulpi_clk) не стробируется (gated off), команда не будет завершена.	0x0	R/W
[7]	PHYSEL	Выбор высокоскоростной PHY USB 2.0 или последовательного полноскоростного передатчика USB 1.1 (USB 2.0 High-Speed PHY or USB 1.1 Full-Speed Serial Transceiver Select). Приложение использует этот бит для выбора между высокоскоростной PHY или полноскоростным передатчиком. 1'b0: высокоскоростной интерфейс UTMI+ USB 2.0 или ULPI PHY. Этот бит всегда имеет значение «0» и доступ только на запись; 1'b1: полноскоростной последовательный передатчик USB 1.1. Этот бит всегда имеет значение «1» и доступ только на запись. Если в coreConsultant выбраны оба типа интерфейсов (значения параметров ненулевые), приложение использует этот бит для того, чтобы выбрать тот интерфейс, который активен в данный	0x0	W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		момент, и в этом случае доступ к нему есть и на чтение, и на запись. Замечание: полноскоростной последовательный передатчик USB 1.1 не поддерживается. Этот бит следует всегда читать как 1'b0.		
[8]	ENBLSLPM	Подключение utmi_sleep_n и utmi_l1_suspend_n (EnblSlpM). Приложение использует этот бит для управления установкой utmi_sleep_n и utmi_l1_suspend_n для PHY в режиме L1. 1'b0: установка utmi_sleep_n и utmi_l1_suspend_n в системе не передается к внешней PHY; 1'b1: установка utmi_sleep_n и utmi_l1_suspend_n передается из системы к внешней PHY. Замечание: - если используется PHY от Synopsys, необходимо устанавливать в этот бит «1» для Port0; - в режиме “Device”, перед отправкой любой команды конечным точкам устройства при работе на скоростях 2.0, следует отключить этот бит и переподключить его после завершения команды. Если, когда устройство находится в режиме L2, отправка команды производится без отключения этого бита, и если mac2_clk (utmi_clk/ulpi_clk) не стробируется (gated off), команда не будет завершена.	0x0	R/W
[9]	XCVRDLY	Задержка передатчика (Transceiver Delay). Подключает задержку между установкой сигнала выбора передатчика UTMI/ULPI (для высокоскоростного (HS) режима) и установкой сигнала TxValid в процессе высокоскоростного чирпирования (HS Chirp). Когда в этот бит установлена «1», то, для отправки chirp-K, между моментом, когда в поле выбора передатчика установлено 2'b00 (высокоскоростной режим) и моментом, когда в сигнал TxValid установлен 0, добавляется задержка (длительностью примерно 2,5 мкс). Эта задержка требуется для некоторых UTMI/ULPI PHY. Замечание: - если функция гибернации подключена, когда система устройства выходит из состояния отключенного питания, необходимо переинициализировать этот бит соответствующим значением, потому что система не сохраняет и не восстанавливает значение этого бита во время гибернации. - этот бит валиден только в режиме “Device”.	0x0	R/W
[13:10]	USBTRDTIM	Длительность цикла оборота USB 2.0 (USB 2.0 Turnaround Time), USBTrdTim.	0x9	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Устанавливает длительность цикла оборота в тактах синхросигнала PHY. Определяет время ответа на MAC-запрос к контроллеру буфера пакетов (Packet FIFO Controller, PFC) для вызова данных из DFIFO (SPRAM). Ниже указаны требуемые значения для минимальной частоты шины SoC в 60 МГц. Длительность цикла оборота USB является критическим критерием сертификации при использовании длинных кабелей и уровней в пять концентраторов (five hub levels). Требуемые значения для этого поля: 4'h5: когда MAC-интерфейс - 16-битный UTMI+. 4'h9: когда MAC-интерфейс - 8-битный UTMI+/ULPI. Если синхросигнал шины SoC имеет частоту ниже 60 МГц, и длительность цикла оборота USB не приближается к критическим значениям, в это поле может быть установлено большее значение. Замечание: это поле валидно только в режиме “Device”.		
[14]	reserved_14	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0x1. Ограничение записи: writeAsRead.	0x0	R
[15]	ULPIAUTORES	Авто-возобновление ULPI (ULPI Auto Resume). Устанавливает бит авто-возобновления в регистре управления интерфейсом в ULPI PHY. 1'b0: в PHY не используется функция авто-возобновления. 1'b1: в PHY используется функция авто-возобновления. Это поле валидно только когда параметр RTL DWC_USB3_HSPHY_INTERFACE = 2 или 3.	0x0	R/W
[16]	reserved_16	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0x1. Ограничение записи: writeAsRead.	0x0	R
[17]	ULPIEXTVBUSDRV	Внешний привод ULPI VBUS (ULPI External VBUS Drive). Выбирает источник питания для подачи 5В на VBUS в интерфейсе ULPI PHY. 1'b0: PHY подает питание на VBUS через внутренний генератор подкачки заряда (по умолчанию). 1'b1: PHY подает питание на VBUS через внешний источник питания. (Только когда параметр RTL DWC_USB3_HSPHY_INTERFACE = 2 или 3).	0x0	R/W
[18]	ULPIEXTVBUSINDIACTOR	Внешний индикатор ULPI VBUS (ULPI External VBUS Indicator), ULPIExtVbusIndicator.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Указывает на индикатор максимального тока (over-current indicator) ULPI PHY VBUS. 1'b0: PHY использует внутренний компаратор валидации (valid comparator) VBUS. 1'b1: PHY использует внешний компаратор валидации VBUS. Это поле валидно только когда параметр RTL DWC_USB3_HSPHY_INTERFACE = 2 или 3.		
[21:19]	LSIPD	Межпакетное время в низкоскоростном режиме (LS Inter-Packet Time). Это поле указывает длительность паузы между пакетами Tx-to-Tx для низкоскоростных (LS) устройств. Кодирование значений: 0: 2 интервала передачи бита; 1: 2.5 интервала передачи бита; 2: 3 интервала передачи бита; 3: 3.5 интервала передачи бита; 4: 4 интервала передачи бита; 5: 4.5 интервала передачи бита; 6: 5 интервалов передачи бита; 7: 5.5 интервалов передачи бита; Замечания: - это поле применимо только в режиме “Host”. - для нормальной работы (работы с большинством низкоскоростных устройств) следует установить для этого поля значение по умолчанию 3'h2 (3 интервала передачи бита). - возможности программировать длительность межпакетной паузы в низкоскоростных устройствах и длительность цикла оборота предоставлены для поддержки некоторых устаревших низкоскоростных устройств, которые могут требовать иных задержек, чем фиксированные или задержки по умолчанию. Например, низкоскоростная мышь AOpen требует для исправной работы длительность межпакетной паузы, равную 3 интервалам передачи бита. - при программировании значений LSIPD/LSTRDTIM следует включать длительности задержек PHY. Например, если задержка TxEndDelay используемой PHY в низкоскоростном режиме - 30 тактов синхросигнала UTMI/ULPI CLK, тогда следует вычесть эту задержку (~1 интервал передачи бита в низкоскоростном режиме (LS bit time)) из требований к задержке устройства.	0x2	R/W
[24:22]	LSTRD	Длительность цикла оборота в низкоскоростном режиме (LS Turnaround Time), LSTRDTIM. Это поле указывает длительность межпакетной паузы Rx-to-Tx в низкоскоростных (LS) устройствах. Кодирование значений: 0: 2 интервала передачи бита; 1: 2.5 интервала передачи бита; 2: 3 интервала передачи бита;	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		3: 3.5 интервала передачи бита; 4: 4 интервала передачи бита; 5: 4.5 интервала передачи бита; 6: 5 интервала передачи бита; 7: 5.5 интервала передачи бита. Замечания: - это поле применимо только в режиме “Host”. - для нормальной работы (работы с большинством низкоскоростных устройств) следует установить для этого поля значение по умолчанию 3'h0 (2 интервала передачи бита). - возможности программировать длительность межпакетной паузы в низкоскоростных устройствах и длительность цикла оборота предоставлены для поддержки некоторых устаревших низкоскоростных устройств, которые могут требовать иных задержек, чем фиксированные или задержки по умолчанию. Например, низкоскоростная мышь Open требует для исправной работы длительность межпакетной паузы, равную 3 интервалам передачи бита. - при программировании значений LSIPD/LSTRDTIM следует учитывать длительности задержек PHY. Например, если задержка TxEndDelay используемой PHY в низкоскоростном режиме - 30 тактов синхросигнала UTMI/ULPI CLK, тогда следует вычесть эту задержку (~1 интервал передачи бита в низкоскоростном режиме (LS bit time)) из требований к задержке устройства.		
[25]	reserved_25	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0x1. Ограничения записи: writeAsRead.	0x0	R
[26]	INV_SEL_HSIC	Драйвер приложения использует этот бит для управления функцией подключения/отключения HSIC (High Speed Inter-Chip). Когда в этот бит установлена «1», этот бит замещает и функционально инвертирует входной сигнал if_select_hsic. Если {INV_SEL_HSIC, if_select_hsic} имеет значение: 00: возможность HSIC отключена; 01: возможность HSIC подключена; 10: возможность HSIC подключена; 11: возможность HSIC отключена. Если контроллер работает как контроллер без возможностей HSIC (non-HSIC-capable), он может подключаться только к PHY, не имеющим возможности HSIC. Если он работает как контроллер с подключенными возможностями HSIC (HSIC-capable), он может подключаться к PHY, имеющим возможности HSIC. Если параметр DWC_USB3_ENABLE_HSIC имеет значение «0», этот бит зарезервирован.	0x0	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
		При выборе функции HSIC, следует перевести первой в режим HSIC сторону, выполняющую роль хоста, а затем переводить в режим HSIC сторону, выполняющую роль устройства. Если сторона, выполняющая роль устройства, переведена в режим HSIC первой, и если хост не видит соединения в этом режиме, следует отключить режим HSIC для устройства и выбрать его вновь через настройку if_select_hsic или бит регистра GUSB2PHYCFGn[26], чтобы гарантировать, что устройство может подсоединяться к хосту. Ограничение записи: writeAsRead.		
[28:27]	HSIC_CON_WIDTH_ADJ	Этот бит используется в режиме “Device” с подключением интерфейса HSIC. По умолчанию, длительность соединения для контроллера прибора HSIC равна трем периодам строба (strobe period). Можно изменить это значение на 4, 5 или 6 периодов строба, если установить в это поле значения 1, 2 или 3, соответственно. Это значение добавляется к длительности соединения по умолчанию. Ограничение записи: writeAsRead.	0x0	R
[29]	ULPI_LPM_WITH_OPMODE_CHK	Поддержка управления питанием каналов (LPM) для ULPI без передачи токена NOPID к ULPI PHY. Если этот бит установлен, ожидается, что ULPI PHY уточнит идентификатор EXT PID значением OPMODE=2'b00 для LPM и не будет использовать его как NOPID. Чтобы уточнить эти особенности поведения PHY, следует обращаться к поставщику используемой PHY. Этот бит валиден только когда параметр DWC_USB3_HSPHY_INTERFACE равен 2 или 3. 1'b0: перед отправкой EXTPID для LPM предварительно производится отправка NOPID; 1'b1: отправка EXTPID производится без предварительной отправки NOPID. Замечание: этот бит валиден только в режиме “Host”. Для PHY от Synopsys этот бит должен иметь значение «0».	0x0	R/W
[30]	U2_FREECLK_EXISTS	Определяет, обеспечивает ли PHY в USB 2.0 автономный синхросигнал PHY, который активен, когда активен ввод управления синхросигналом. Если используемая в USB 2.0 PHY обеспечивает автономный синхросигнал PHY, он должен быть соединен со вводом utmi_clk[0]. Остающиеся utmi_clk[n] должны быть подсоединены к соответствующим синхросигналам портов. Система использует синхросигнал порта Port0 для генерации внутреннего синхросигнала mac2. 1'b0: автономного сигнала в USB 2.0 нет; 1'b1: автономный сигнал в USB 2.0 есть.	0x1	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Замечание: когда система конфигурируется только для работы с устройством (DWC_USB3_MODE = 0), не следует устанавливать «1» в этот бит.		
[31]	PHYSOFTRST	Мягкий сброс PHY UTMI (UTMI PHY Soft Reset). Запускает установку сигнала usb2phy_reset для сброса PHY UTMI. Неприменимо к ULPI, поскольку ULPI PHY сбрасываются через регистр FunctionControl.Reset, и система автоматически делает запись в этот регистр при сбросе системы (vcc_reset_n, USBCMD.HCRST, DCTL.SoftReset, или GCTL.SoftReset).	0x0	R/W

26.3.40 Зарезервированный регистр (GUSB2I2CCTL(#n)(for n == 0))

Приложение использует этот регистр, чтобы обращаться к устройствам типа OTG (On-The-Go - подключение «на ходу»), подсоединенным к системе OTG через интерфейс I2C. Он реализуется только если во время конфигурирования coreConsultant выбрана опция “Enable I2C Interface” («Подключить интерфейс I2C») (параметр DWC_USB3_I2C_INTERFACE = 1).

Интерфейс I2C в системе OTG может читать и писать в полях регистров подсоединенного устройства I2C. Таблица ниже описывает поля этого регистра.

В конфигурациях, использующих только устройство (Device-only), необходим только один регистр. В режиме “Host” реализуются регистры для каждого порта.

Смещение: 0xc240 + (i * 0x4), где i = 0..=[eval_param DWC_USB3_NUM_U2_ROOT_PORTS]-1.

Поля регистра GUSB2I2CCTL(#n)(for n == 0) описаны в Таблица 26.50.

Таблица 26.50. Зарезервированный регистр

Диапазон	Название	Описание	Исходное значение	Тип доступа
[31:0]	reserved_31_0	Зарезервировано для дальнейшего использования. Контролепригодность: writeAsRead. Маска сброса: 0xffffffff. Ограничение записи: writeAsRead.	0x0	R

26.3.41 Глобальный регистр управления USB 2.0 ULPI PHY от поставщика (GUSB2PHYACC_ULPI(#n)(for n == 0))

Приложение использует этот регистр для обращения к регистрам PHY. Этот регистр всегда реализован, когда интерфейс PHY ULPI подключен в процессе конфигурирования coreConsultant (параметр DWC_USB3_HSPHY_INTERFACE == 2 или DWC_USB3_HSPHY_INTERFACE == 3).

Для ULPI PHY, система использует интерфейс ULPI для обращения к регистрам PHY.

Приложение устанавливает регистр управления от поставщика для обращения к регистрам PHY и задает время обращения к регистрам PHY. Приложение опрашивает (polls) бит “VStatus Done” в этом регистре для завершения обращения к регистру PHY.

В конфигурациях “Device-only” (только устройство) необходим только один регистр. В режиме “Host” реализуется по регистру для каждого порта.

Смещение: 0xc280 + (i * 0x4), где i = 0..=[eval_param DWC_USB3_NUM_U2_ROOT_PORTS]-1.

Поля регистра GUSB2PHYACC_ULPI(#n)(for n == 0) описаны в Таблица 26.51.

Таблица 26.51. Глобальный регистр управления USB 2.0 ULPI PHY от поставщика

Диапазон	Название	Описание	Исходное значение	Тип доступа
[7:0]	REGDATA	Контролепригодность: нетестируемо. Ограничение записи: writeAsRead.	0x0	R
[15:8]	EXTREGADDR	Контролепригодность: нетестируемо. Ограничение записи: writeAsRead.	0x0	R
[21:16]	REGADDR	Адрес регистра. 6-битный адрес регистра PHY для немедленного обращения к набору регистров PHY. Для расширенного обращения к набору регистров PHY в этот бит следует установить 6'h2F. Замечание: эти биты применимы только для ULPI. Контролепригодность: нетестируемо. Ограничение записи: writeAsRead.	0x0	R
[22]	REGWR	Запись в регистр. Приложение устанавливает этот бит для записей в регистр и очищает его для чтений из регистра. Замечание: этот бит применим только для обращений на запись/чтение к регистру ULPI. Контролепригодность: нетестируемо. Ограничение записи: writeAsRead.	0x0	R
[23]	VSTSBSY	Контролепригодность: нетестируемо. Ограничение записи: writeAsRead.	0x0	R
[24]	VSTSDONE	Контролепригодность: нетестируемо. Изменчивость: true.	0x0	R
[25]	NEWREGREQ	Запрос нового регистра (New Register Request). Приложение устанавливает этот бит для нового обращения к управлению от поставщика (vendor control access). Установка 1 в этот бит устанавливает utmi_vcontrolload_n (1'b0) в интерфейсе UTMI. Контролепригодность: нетестируемо.	0x0	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Изменчивость: true. Ограничение записи: writeAsRead.		
[26]	DISUIPIDRVR	Контролепригодность: нетестируемо. Изменчивость: true. Ограничение записи: writeAsRead.	0x0	R
[31:27]	reserved_31_27	Зарезервировано. Контролепригодность: нетестируемо. Маска сброса: 0x1f. Ограничение записи: writeAsRead.	0x0	R

26.3.42 Глобальный регистр управления USB 2.0 UTMI PHY от поставщика (GUSB2PHYACC_UTMI(#n)(for n == 0))

Приложение использует этот регистр для обращения к регистрам PHY. Он реализуется только если интерфейс управления PHY от поставщика (PHY Vendor Control Interface) или интерфейс PHY ULPI был выбран в процессе конфигурирования в coreConsultant (параметр DWC_USB3_VENDOR_CTL_INTERFACE == 1 или DWC_USB3_HSPHY_INTERFACE == 2 или DWC_USB3_HSPHY_INTERFACE == 3).

В режиме работы UTMI, функциональность этого регистра может быть использована только если подключен интерфейс управления UTMI от поставщика (UTMI Vendor Control Interface).

Для UTMI+ PHY, контроллер использует интерфейс управления UTMI+ от поставщика (UTMI+ Vendor Control Interface) для обращения к регистрам PHY. Для UTMI+ нет дифференциации на запись/чтение. Там есть только обращение к управлению от поставщика. Приложение устанавливает регистр управления от поставщика для обращения к регистрам PHY и задает время обращения к регистру PHY. Приложение опрашивает бит “VStatus Done” в этом регистре для завершения обращения к регистру PHY.

В конфигурациях “Device-only” (только устройство) необходим только один регистр. В режиме “Host” реализуется по регистру для каждого порта.

Смещение: 0xc280 + (i * 0x4), где i = 0..=[eval_param DWC_USB3_NUM_U2_ROOT_PORTS]-1.

Поля регистра GUSB2PHYACC_UTMI(#n)(for n == 0) описаны в Таблица 26.52.

Таблица 26.52. Глобальный регистр управления USB 2.0 UTMI PHY от поставщика

Диапазон	Название	Описание	Исходное значение	Тип доступа
[7:0]	REGDATA	Контролепригодность: нетестируемо. Ограничение записи: writeAsRead.	0x0	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
[15:8]	VCTRL	Это поле содержит 4-битный адрес регистра и заданный поставщиком 4-битный параллельный вывод шины. Биты [11:8] этого поля также помещаются на биты [3:0] сигнала вывода utmi_vcontrol. Контролепригодность: нетестируемо. Ограничение записи: writeAsRead.	0x0	R
[21:16]	REGADDR	Контролепригодность: нетестируемо. Ограничение записи: writeAsRead.	0x0	R
[22]	REGWR	Контролепригодность: нетестируемо. Ограничение записи: writeAsRead.	0x0	R
[23]	VSTSBSY	Контролепригодность: нетестируемо. Ограничение записи: writeAsRead.	0x0	R
[24]	VSTSDONE	Контролепригодность: нетестируемо. Изменчивость: true.	0x0	R
[25]	NEWREGREQ	Контролепригодность: нетестируемо. Ограничение записи: writeAsRead.	0x0	R
[26]	DISUIPIDRVR	Контролепригодность: нетестируемо. Ограничение записи: writeAsRead.	0x0	R
[31:27]	reserved_31_27	Зарезервировано. Контролепригодность: нетестируемо. Маска сброса: 0x1f. Ограничение записи: writeAsRead.	0x0	R

26.3.43 Глобальный регистр настроек PIPE USB 3.0 (GUSB3PIPECTL(#n) (for n = 0; n <= DWC_USB3_NUM_U3_ROOT_PORTS-1))

Приложение использует этот регистр для конфигурирования USB3 PHY и интерфейса PIPE.

В конфигурациях, относящихся только к устройству, требуется только один регистр.

В режиме “Host” реализуется по регистру для каждого порта.

Подробнее биты GUSB3PIPECTL(#n) описаны в Разделе «Регистр GUSB3PIPECTL(#n)» в Руководстве пользователя по сверхскоростному контроллеру USB 3.0 DWC.

Замечание:

регистры GUSB3PIPECTLn неприменимы в режиме, использующем только USB 2.0.

Смещение: 0xc2c0 + (i * 0x4), где i = 0..=[DWC_USB3_NUM_U3_ROOT_PORTS]-1.

Поля регистра GUSB3PIPECTL(#n) (for n = 0; n <= DWC_USB3_NUM_U3_ROOT_PORTS-1) описаны в Таблица 26.53.

Таблица 26.53. Глобальный регистр настроек PIPE USB 3.0

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	ELASTIC_BUFFER_MODE	Режим эластичного буфера (Elastic Buffer Mode). Подробнее см. таблица 5-3 спецификации PIPE3.	0x0	R/W
[2:1]	SS_TX_DE_EMPHASIS	Коррекция предыскажений Tx (Tx Deemphasis). Значение, подаваемое к PHY, управляется машиной настройки каналов и состояний статусов (Link Training and Status State Machine, LTSSM) во время режима соответствия (Compliance mode) USB3. Подробнее см. Таблица 5-3 спецификации PIPE3.	0x1	R/W
[5:3]	TX_MARGIN	Допуски Tx [2:0] (Tx Margin[2:0]). Подробнее см. Таблица 5-3 спецификации PIPE3.	0x0	R/W
[6]	TX_SWING	Максимальное отклонение Tx (Tx Swing), TxSwing. Подробнее см. спецификацию PIPE3.	0x0	R/W
[7]	SSICEn	Подключение SSIC USB3 (USB3 SSIC Enable). Этот бит валиден только когда параметр DWC_USB3_EN_SSIC=1 в coreConsultant, в противном случае в этот бит необходимо установить 1'b0. - 1'b0: активен интерфейс Pipe. Интерфейс RMMI на соответствующем порте сбрасывается системой. - 1'b1: активен интерфейс RMMI на соответствующем порте, соединенном с M-PHY, а интерфейс pipe неактивен. Перед обращением к соответствующему регистру GUSB3RMMICTLn необходимо установить этот бит. Замечание: когда в параметр DWC_USB3_NUM_SSIC_PORTS установлено нулевое значение, тип доступа этого бита регистра - R/W, но регистр нефункционален.	0x0	R/W
[8]	RX_DETECT_to_Polling_L FPS_Control	Настройка RX_DETECT для Polling.LFPS (RX_DETECT to Polling.LFPS Control). 1'b0 (по умолчанию): подключает задержку в 400мкс для запуска упорядочивания (Polling) LFPS после RX_DETECT. Это позволяет установиться смещению VCM (common-mode voltage, синфазное напряжение) на подходящем уровне. 1'b1: отключает задержку в 400мкс для запуска упорядочивания LFPS после RX_DETECT. В процессе сертификации контроллера для сторонней PHY наблюдается, что PHY не	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		удовлетворяет активному напряжению стандартного режима AC Tx (VTX-CM-ACPP_ACTIVE <100мВ), если канал начинает упорядочивание в пределах 80 мкс с момента исполнения rx.detect. Для удовлетворения спецификации VTX-CM-ACPP_ACTIVE, упорядочивание должно производиться с большей задержкой. Если используемая PHY не обладает данной проблемой, можно установить в этот бит «1», что позволит упорядочиванию начинаться в пределах 80 мкс.		
[9]	LFPSFILTER	Фильтр LFPS (LFPS Filter, LFPSFilt). Когда установлен, фильтрует получение LFPS, когда pipe3_RxValid находится в режиме питания PHY P0. Таким образом, следует игнорировать получение LFPS от PHY, если только pipe3_Rxelecidle и pipe3_RxValid не деактивированы одновременно.	0x0	R/W
[10]	P3ExSigP2	Сигнал выхода из P3 в P2 (P3 Exit Signal in P2). Когда этот бит установлен, система всегда меняет режим питания PHY на P2 перед тем, как пытаться провести процедуру согласования выхода из U3 (U3 exit handshake). Этот бит используется только для некоторых сторонних PHY (не от Synopsys), которые не могут создать LFPS в режиме P3. Замечание: этот бит используется для работы со сторонней сверхскоростной PHY. Для работы с PHY от Synopsys в него необходимо устанавливать «0».	0x0	R/W
[11]	P3P2TranOK	Успешные переключения P3-P2 (P3 P2 Transitions OK). Когда установлен, система напрямую переключается из режима питания Phy P2 в P3 или из режима P3 в P2. Когда не установлен, в процессе переключения всегда производится вход в P0 как в промежуточное состояние во время переключений между P2 и P3, как указано в спецификации PIPE3. В соответствии со спецификацией PIPE3, любое прямое переключение между P3 и P2 недопустимо. Замечание: этот бит используется для работы со сторонней сверхскоростной PHY. Для работы с PHY от Synopsys в него необходимо устанавливать «0».	0x0	R/W
[12]	LFPSP0Algn	Ориентирование LFPS в режиме P0 (LFPS P0 Align).	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Когда этот бит установлен: - система деактивирует переключение LFPS на фронте такта синхросигнала, на котором следует запрос о переходе в режим питания Phy P0 при выходе из U1, U2, или U3 при низком питании. В противном случае, переключение LFPS активируется на один такт раньше. - система запрашивает переключение идентификатора через два периода pipe3_rx_pclks после того, как PHY активирует PhyStatus в результате переключения из P1 или P2 в режим P0. В нынешний момент этот бит применяется только в USB 3.0 HUB вместе с PHY производства Synopsys. Для других систем USB 3.0 – “Host”, “Device” и “DRD”, этот бит не требуется.		
[13]	SkipRxDet	Пропуск обнаружения Rx (Skip Rx Detect). Когда установлен, система пропускает Rx Detection, если pipe3_RxElecIdle низок. Пропуск определяется как ожидание до соответствующего таймаута, затем - повтор действия.	0x0	R/W
[14]	AbortRxDetInU2	Аварийное прекращение Rx Detect в U2 (Abort Rx Detect in U2). Когда этот бит установлен, и канал находится в режиме U2, система прервёт обнаружение приемника, если получит сигнал LFPS выхода из U2 от удаленного партнера по каналу. Этот бит предназначен только для нисходящих портов. Замечание: этот бит используется для работы со сторонней сверхскоростной PHY. Для работы с PHY от Synopsys в него необходимо устанавливать «0».	0x0	R/W
[16:15]	DATWIDTH	Разрядность данных PIPE (PIPE Data Width). 2'b00: 32 бита; 2'b01: 16 бит; 2'b10: 8 бит. Спустя один такт после сброса, эти биты получают значения, видимые в pipe3_DataBusWidth. В наборе тестов для симуляции используется параметр из coreConsultant для конфигурирования VIP. Эти биты в параметре coreConsultant должны связывать разрядность данных используемой PHY и порта pipe3_DataBusWidth. Контролепригодность: нетестируемо. Изменчивость: true.	0x0	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Ограничение записи: writeAsRead.		
[17]	SUSPENDENABLE	Приостановка сверхскоростной PHY USB3.0 (Suspend USB3.0 SS PHY), Suspend_en. Когда установлен, и если условия приостановки валидны, PHY USB 3.0 входит в режим приостановки. Для конфигураций DRD/OTG рекомендуется устанавливать «0» в этот бит во время конфигурирования в coreConsultant. Если в этот бит установлена «1», тогда, после сброса при запуске, его должно обнулить приложение. Также необходимо, чтобы приложение установило в него «1» после завершения инициализации системы. Для всех остальных конфигураций в процессе конфигурирования системы в этот бит можно устанавливать «1».	0x0	R/W
[18]	DELAYP1TRANS	Задержка смены режима питания PHY с P0 на P1/P2/P3 при изменении режима канала с U0 на U1/U2/U3, соответственно. 1'b1: при входе в U1/U2/U3 задерживает переход в P1/P2/P3 до тех пор, пока не будет получен сигнал pipe3, Pipe3_RxElecIdle не получит значение «1», а pipe3_RxValid – «0»; 1'b0: при входе в U1/U2/U3 переходит в P1/P2/P3 без проверки значений Pipe3_RxElecIdle и pipe3_RxValid. Замечание: - если используется PHY от Synopsys, следует связаться с пользовательской поддержкой Synopsys для получения рекомендаций по установке этого бита, поскольку он зависит от узла (node dependent). - если используется сторонняя сверхскоростная PHY, следует связаться с поставщиком PHY для получения рекомендаций по установке этого бита.	0x1	R/W
[21:19]	DelayP1P2P3	Задержка P1P2P3 (Delay P1P2P3). Задерживает запрос на переход из режима питания P0 в P1/P2/P3 при входе в U1/U2/U3 до тех пор, пока не возникнет ошибка (DWC_USB3_GUSB3PIPECTL_INIT[21:19] *8) 8B10B, или Pipe3_RxValid упадет до 0. Для подключения этого функционала DWC_USB3_GUSB3PIPECTL_INIT[18] должен иметь значение «1».	0x0	R/W
[22]	DisRxDetU3RxDet	Отключение обнаружения приемника в U3/Rx.Det (Disable Receiver Detection in U3/Rx.Det).	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Когда установлен, система не занимается обработкой обнаружения приемника ни в режиме U3, ни в Rx.Detect. Необходимо воспользоваться битом DWC_USB3_GUSB3PIPECTL_INIT[23] для запуска обнаружения приемника вручную. Этот бит может использоваться только для нисходящего порта. Для восходящих портов в этот бит необходимо устанавливать «0». Эта функция не должна быть подключена в стандартном режиме работы. Если необходимо использовать эту функцию, следует связаться с Synopsys.		
[23]	StartRxDetU3RxDet	Запуск обнаружения приемника в U3/Rx.Detect (Start Receiver Detection in U3/Rx.Detect). Если установлен бит DWC_USB3_GUSB3PIPECTL_INIT[22], и канал находится либо в режиме U3, либо в Rx.Detect, система начинает обнаружение приемника на верхнем фронте сигнала этого бита. Это может использоваться только для нисходящих портов. Для восходящих портов в этот бит необходимо устанавливать «0». Эта функция не должна быть подключена для стандартной работы. Если есть необходимость воспользоваться этой функцией, следует связаться с Synopsys. Контролепригодность: writeAsRead.	0x0	W
[24]	request_p1p2p3	Всегда запрашивать режим питания P1/P2/P3 для U1/U2/U3 (Always Request P1/P2/P3 for U1/U2/U3). Когда установлен, система всегда запрашивает изменение режима питания для PHY с P0 на P1/P2/P3 при переходе из U0 в U1/U2/U3. Если в этом бите «0», и происходит немедленный выход из Ux (инициированный удаленно или локально), система не запрашивает изменение режима питания на P1/P2/P3. Замечание: для PHY от Synopsys в этот бит следует устанавливать «1». Для сторонних сверхскоростных PHY следует проконсультироваться у поставщика PHY.	0x1	R/W
[25]	u1u2exitfail_to_recov	Сбой возобновления при выходе из U1U2 (U1U2exitfail to Recovery). Когда этот бит установлен, и происходит сбой процедуры согласования (handshake) LFPS в U1/U2, машина состояний формирования соединений и статуса (LTSSM) переходит из U1/U2 в состояние возобновления вместо того, чтоб	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		переходить в SS.Inactive. Если происходит сбой возобновления, тогда LTSSM может войти в SS.Inactive. Эта функция применяется только для оптимизации. Она предохраняет от проблем совместимости, если удаленный канал не производит согласование надлежащим образом.		
[26]	ping_enhancement_en	Подключение усовершенствования пинга (Ping Enhancement Enable). Когда установлен, длительность таймаута получения пинга на нисходящем (Downstream) порту U1 становится 500 мс вместо 300 мс. Минимальная длительность получения пинга LFPS - 8 нс (один такт mac3_clk). Это поле валидно только для нисходящего порта. Замечание: этот бит используется для работы со сторонней сверхскоростной PHY. Для работы с PHY от Synopsys в него необходимо устанавливать «0».	0x0	R/W
[27]	Ux_exit_in_Px	Выход из Ux в режиме питания Px (Ux Exit in Px). 0: (по умолчанию) система производит выход из состояния U1/U2/U3 в режиме питания PHY P0. 1: система производит выход из состояния U1/U2/U3 в режиме питания PHY P1/P2/P3, соответственно. Замечание: этот бит используется для работы со сторонней сверхскоростной PHY. Для работы с PHY от Synopsys в него необходимо устанавливать «0».	0x0	R/W
[28]	DisRxDetP3	Обнаружение отключенного приемника в P3 (Disabled receiver detection in P3). Если PHY находится в режиме P3 и системе необходимо произвести обнаружение приемника: 0: (по умолчанию) система производит обнаружение приемника в режиме P3; 1: система меняет режим питания PHY на P2 и затем производит обнаружение приемника. После обнаружения приемника, система меняет режим питания обратно на P3.	0x0	R/W
[29]	U2P3ok	Успешное переключение в P3 для U2 (P3 OK for U2). 0: по умолчанию: во время режима U2/SS.Inactive на канале, помещает PHY в состояние P2. 1: во время режима U2/SS.Inactive на канале, помещает PHY в состояние P3.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Замечание: для порта, если GUSB3PIPECTL[7]=1 и GUSB3PIPECTL[29]=1, следует установить GUSB3PIPECTL[11]=1.		
[30]	HstPrtCmpl	Эта функция тестирует паттерны соответствия (compliance patterns) PIPE PHY, что избавляет от нужды в дополнительном тестовом приспособлении на кабеле USB 3.0. Этот бит разрешает поместить канал сверхскоростного порта в режим соответствия (compliance state). По умолчанию, в этот бит следует устанавливать 1'b0. В лабораторном тестировании соответствия, канал сверхскоростного порта входит в режим соответствия после провала первой последовательности упорядочивания (polling sequence) после подключения. Во время запуска тестов на соответствие в этот бит следует устанавливать «0». Последовательность пользования этим функционалом следующая. 1. Отключить любые подключенные устройства. 2. Произвести USBCMD.HCRST или сброс питания на чипе (power-on-chip reset). 3. Установить PORTSC.PLS=0xA. 4. Установить PORTSC.PP=0. 5. Установить GUSB3PIPECTL. HstPrtCmpl=1. Это помещает канал в режим соответствия. Для сдвига паттерна соответствия следует исполнить следующую последовательность (переключение установленного GUSB3PIPECTL. HstPrtCmpl). 1. Установить GUSB3PIPECTL.HstPrtCmpl=0. 2. Установить GUSB3PIPECTL.HstPrtCmpl=1. Это сдвигает канал к следующему паттерну соответствия. Для выхода из режима соответствия следует произвести USBCMD.HCRST или сброс питания на чипе.	0x0	R/W
[31]	PHYSoftRst	Мягкий сброс PHY USB3 (USB3 PHY Soft Reset). После установки «1» в этот бит, программное обеспечение обязано очистить его. Более подробно работа с этим битом представлена на рисунке «Программные сбросы, упорядочивание и требования к синхросигналам PHY» в справочнике по	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		сверхскоростному контроллеру USB 3.0 DWC. Контролепригодность: writeAsRead.		

26.3.44 Глобальный регистр размера буфера FIFO на передачу (GTXFIFOSIZ(#n)(for n = 0; n <= 31))

Этот регистр определяет адрес начала памяти RAM и глубину (оба значения - в словах MDWIDTH-bit) для каждого реализованного буфера TxFIFO. Количество буферов TxFIFO зависит от параметров конфигурации, включая количество конечных точек IN устройства, количество экземпляров шины хоста (Host Bus Instances) и присутствие возможности отладки.

Значения по умолчанию данного регистра для каждого режима присваиваются в coreConsultant на основе максимального размера пакета, количества пакетов, которые необходимо буферизовать, скорости экземпляра шины хоста, запаздывания шины, и режима работы (“Host”, “Device” или DBC (возможность отладки)). При сбросе и переключениях режимов, аппаратное обеспечение автоматически программирует в эти регистры значения по умолчанию.

Соответственно, как правило, нет необходимости в том, чтобы программное обеспечение изменяло предустановленные значения по умолчанию.

Для режима возможности отладки, поставленные в соответствие в настоящий момент номера EP0 IN (конечной точки 0 IN) и EP1 IN для TxFIFO могут быть прочтены из регистра GFIFOPRIDBC.

Для режима работы OTG (On-The-Go, на «ходу»), когда система переключается в режим “Host”, следует программировать правильное значение в регистр GTXFIFOSIZ только после того, как в OCTL.PeriMode запрограммировано значение 1'b0.

Подробнее применение регистров GTXFIFOSIZn и GRXFIFOSIZn для разных режимов работы описано в главе «Детали архитекутры» справочника.

Смещение: 0xc300 + (i * 0x40), где i = 0..0.

Поля регистра GTXFIFOSIZ(#n)(for n = 0; n <= 31) описаны в Таблица 26.54.

Таблица 26.54. Глобальный регистр размера буфера FIFO на передачу

Диапазон	Название	Описание	Исходное значение	Тип доступа
[15:0]	TXFDEP_N	Глубина TxFIFO (TxFIFO Depth). Это поле содержит глубину TxFIFOn в битовых словах MDWIDTH-bit. Минимальное значение: 32. Максимальное значение: 32,768. Подробнее это поле описано в главе «Интеграция системы» (“Integrating the Core”) Руководства пользователя по сверхскоростному контроллеру USB 3.0 DWC.	0x42	R/W
[31:16]	TXFSTADDR_N	Адрес начала памяти RAM для передач FIFOn (Transmit FIFOn RAM Start Address). Это поле содержит адрес начала памяти для TxFIFOn в словах MDWIDTH-bit.	0x0	R/W

26.3.45 Глобальный регистр размера буфера FIFO на получение (GRXFIFOSIZ(#n)(for n = 0; n <= 31))

Этот регистр задает начальный адрес памяти RAM и глубину (оба значения - в битовых словах MDWIDTH) для каждого реализованного буфера RxFIFO. Количество буферов RxFIFO зависит от параметров конфигурации, включая количество экземпляров шины хоста (Host Bus Instances) и присутствие возможности отладки (Debug Capability). Режим “Device” требует только одного RxFIFO.

Значения этого регистра по умолчанию для каждого режима присваиваются в coreConsultant на основе максимального размера пакета, количества пакетов, которые следует сохранить в буфере, скорости экземпляра шины хоста, запаздывания шины, и режима работы (“Host”, “Device” или DBC (возможность отладки)). При переключениях режимов и сбросе, аппаратное обеспечение автоматически программирует значения по умолчанию в эти регистры. Соответственно, как правило, не возникает необходимости, чтобы программное обеспечение модифицировало заранее заданные значения по умолчанию.

В режиме возможности отладки, номер буфера RxFIFO, поставленный в соответствие в текущий момент, может быть прочтён из регистра GFIFOPRIDBC.

Подробнее использование регистров GTXFIFOSIZn и GRXFIFOSIZn в разных режимах работы описано в главе «Детали архитектуры» в справочнике.

Смещение: 0xc380 + (i * 0x40), где i = 0..0.

Поля регистра GRXFIFOSIZ(#n)(for n = 0; n <= 31) описаны в Таблица 26.55.

Таблица 26.55. Глобальный регистр размера буфера FIFO на получение

Диапазон	Название	Описание	Исходное значение	Тип доступа
[15:0]	RXFDEP_N	Глубина RxFIFO (RxFIFO Depth (RxFDep_n)). Это поле содержит глубину RxFIFOn в битовых словах MDWIDTH. Минимальное значение: 32. Максимальное значение: 16,38485.	0x185	R/W
[31:16]	RXFSTADDR_N	Начальный адрес RAM RxFIFOn (RxFIFOn RAM Start Address (RxFStAddr_n)). Это поле содержит начальный адрес памяти для RxFIFOn в битовых словах MDWIDTH.	0x0	R/W

26.3.46 Младший глобальный регистр адреса буфера событий (GEVNTADRLO(#n)(for n == 0))

Альтернативный регистр для регистра GEVNTADRn.

Смещение: 0xc400 + (i * 0x10), где i = 0..=[eval_param DWC_USB3_DEVICE_NUM_INT]-1.

Поля регистра GEVNTADRLO(#n)(for n == 0) описаны в Таблица 26.56.

Таблица 26.56. Младший глобальный регистр адреса буфера событий

Диапазон	Название	Описание	Исходное значение	Тип доступа
[31:0]	EVNTADRLO	Адрес буфера событий (Event Buffer Address). Содержит младшие 32 бита начального адреса внешней памяти для буфера событий. В процессе работы аппаратное обеспечение не обновляет этот адрес. Маска сброса: 0xffffffff. Изменчивость: true.	0	R/W

26.3.47 Старший глобальный регистр адреса буфера событий (GEVNTADRHI(#n)(for n == 0))

Альтернативный регистр для регистра GEVNTADRn.

Смещение: 0xc404 + (i * 0x10), где i = 0..=[eval_param DWC_USB3_DEVICE_NUM_INT]-1.

Поля регистра GEVNTADRHI(#n)(for n == 0) описаны в Таблица 26.57.

Таблица 26.57. Старший глобальный регистр адреса буфера событий

Диапазон	Название	Описание	Исходное значение	Тип доступа
[31:0]	EVNTADRHI	Адрес буфера событий (Event Buffer Address (EvntAdrHi)). Содержит старшие 32 бита начального адреса внешней памяти для буфера событий. В процессе работы аппаратное обеспечение не обновляет этот адрес. Маска сброса: 0xffffffff. Изменчивость: true.	0	R/W

26.3.48 Глобальный регистр размера буфера событий (GEVNTSIZ(#n)(for n == 0))

Этот регистр содержит биты размера буфера событий (Event Buffer Size) и маски прерывания события (Event Interrupt Mask). В процессе инициализации при запуске, программное обеспечение должно инициализировать размер числом байтов, зарезервированных под буфер событий. Маска прерывания события замаскирует прерывание, но события всё еще будут накапливаться в очереди. После конфигурирования, программное обеспечение должно сохранять значение поля «Размер буфера событий», когда меняет поле «Маска прерывания событий».

Смещение: 0xc408 + (i * 0x10), где i = 0..=[eval_param DWC_USB3_DEVICE_NUM_INT]-1.

Поля регистра GEVNTSIZ(#n)(for n == 0) описаны в Таблица 26.58.

Таблица 26.58. Глобальный регистр размера буфера событий

Диапазон	Название	Описание	Исходное значение	Тип доступа
[15:0]	EVENTSIZ	Размер буфера событий в байтах (Event Buffer Size in bytes). Содержит размер буфера событий в байтах. Размер должен быть кратен четырем. Это поле программируется программным обеспечением единственный раз, в процессе инициализации. Минимальный размер буфера событий - 32 байта.	0x0	R/W
[30:16]	reserved_30_16	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0x7fff. Ограничение записи: writeAsRead.	0x0	R
[31]	EVNTINTRPTMASK	Маска прерывания событий (Event Interrupt Mask). Когда в это поле установлена «1», предотвращает генерацию прерывания. Однако, даже когда маска установлена, события накапливаются в очереди.	0x0	R/W

26.3.49 Глобальный регистр счетчика буфера событий (GEVNTCOUNT(#n)(for n == 0))

Этот регистр содержит число валидных байтов в буфере событий (Event Buffer). В процессе инициализации, программное обеспечение должно инициализировать счетчик путем записи «0» в поле “Event Count” («Счетчик событий»). Всякий раз, как аппаратное обеспечение записывает новое событие в буфер событий, оно инкрементирует этот счетчик. Большинство событий имеют размер 4 байта, но некоторые события могут занимать несколько 4-байтовых записей. Когда счетчик больше 0, и удовлетворены условия для модерации прерывания (если они подключены), аппаратное обеспечение вызывает соответствующую линию прерывания (в зависимости от бита EvntIntMask в регистре GEVNTSIZn). В процессе прерывания, программное обеспечение обрабатывает одно или более событий из буфера событий.

После этого программное обеспечение должно записать в поле “Event Count” количество обработанных байтов. Если подключена модерация прерываний (Interrupt Moderation), тогда программное оебспечение должно очистить бит EVNT_HANDLER_BUSY.

Задержки пересечения синхросигналов (Clock crossing delays) могут привести к постоянной установке прерывания после того, как программное обеспечение подтвердит получение последнего события. Таким образом, когда установлена линия прерывания, программное обеспечение обязано прочесть регистр GEVNTCOUNT и обрабатывать события только если GEVNTCOUNT имеет значение больше 0.

Смещение: 0xc40c + (i * 0x10), где i = 0..=[eval_param DWC_USB3_DEVICE_NUM_INT]-1.

Поля регистра GEVNTCOUNT(#n)(for n == 0) описаны в Таблица 26.59.

Таблица 26.59. Глобальный регистр счетчика буфера событий

Диапазон	Название	Описание	Исходное значение	Тип доступа
[15:0]	EVNTCOUNT	Счетчик событий (Event Count (EVNTCount)). При чтении возвращает число валидных событий в буфере событий (в байтах). При записи, программное обеспечение декрементирует счетчик на записанную величину. Когда подключена модерация прерываний (то есть, DEVICE_IMODI!= 0), то, когда на этом регистре происходит первая запись для декрементирования счетчика, линия прерывания снимается. Когда модерация прерываний отключена (то есть, DEVICE_IMODI= 0), линия прерывания продолжает оставаться установленной до тех пор, пока счетчик событий не приобретает значение 0 (поведение без модерации). Маска сброса: 0xffff. Изменчивость: true.	0x0	R/W
[30:16]	reserved_30_16	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0x7fff. Ограничение записи: writeAsRead.	0x0	R
[31]	EVNT_HANDLER_BUSY	Обработчик событий занят (Event Handler Busy). Указание на загруженность обработчика событий программного обеспечение устройства. Контроллер устанавливает этот бит, когда установлена линия прерывания, из-за ожидающих обработки событий. Программное обеспечение очищает этот бит (через 1'b1) когда обработка событий завершена (наряду с обновлением счетчика EVNTCOUNT в этом регистре). Контроллер не вызывает линию прерывания для нового события, пока этот бит не будет очищен. Замечание: когда модерация прерывания отключена (то есть, DEVICE_IMODI = 0), этот бит игнорируется. Маска сброса: 0x1. Изменчивость: true.	0x0	R/W

26.3.50 Глобальный регистр аппаратных параметров 8 (GHWPARAMS8)

Этот регистр содержит опции аппаратной конфигурации, которые могут быть подключены через GUI в coreConsultant.

Описание каждого параметра представлено в главе «Параметры» справочника. Эта информация также доступна в coreConsultant: для этого необходимо кликнуть правой кнопкой мыши по ярлыку параметра и выбрать «Что это» (“What's This”); также можно воспользоваться вкладкой «Помощь».

Замечание: некоторые глобальные аппаратные параметры недоступны для модификации через coreConsultant в настоящий момент. Эти настройки находятся в файле <workspace>/src/DWC_usb3_params.v. Пользователь не должен их изменять.

Поля регистра GHWPARAMS8 описаны в Таблица 26.60.

Таблица 26.60. Глобальный регистр аппаратных параметров 8

Диапазон	Название	Описание	Исходное значение	Тип доступа
[31:0]	ghwparams8_32_0	`DWC_USB3_DCACHE_DEPTH_INFO. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x886	R

26.3.51 Глобальный регистр приоритета DMA для буферов FIFO TX устройства (GTXFIFOPRIDEV)

Этот регистр задает относительный уровень приоритета прямого доступа к памяти (Direct Memory Access, DMA) среди буферов TXFIFO устройства (по одному на каждую конечную точку IN). Каждый бит регистра [n] управляет приоритетом на доступ (1: высокий, 0: низкий) для каждого TXFIFO[n]. Когда множество буферов TXFIFO конкурируют за работу с DMA в одно и то же время (например, множество TXQ содержат запросы TX DMA и в соответствующих им TXFIFO есть свободное место), арбитр TX DMA предоставляет доступ на основе пакетного обмена по следующим правилам:

1. Буферы TXFIFO, имеющие высокий приоритет, получают доступ по циклическому арбитражу.

2. Буферы TXFIFO, имеющие низкий приоритет, получают доступ по циклическому арбитражу только после того, как у буферов TXFIFO с высоким приоритетом больше не остается задач на обработку (либо очереди TXQ опустошены, либо соответствующие им TXFIFO заполнены).

Для пакетов сбора/разброса (scatter-gather), арбитр предоставляет одному и тому же FIFO возможность последовательной отправки запросов к DMA до тех пор, пока не будет завершена работа со всем пакетом.

При конфигурировании периодических конечных точек IN, программное обеспечение обязано устанавливать в [n] бит этого регистра «1». Здесь n соответствует буферу TXFIFO. Таким образом, доступ DMA для изохронных конечных точек IN или конечных точек прерывания IN будет приоритетнее, чем для контрольных конечных точек IN или конечных точек типа bulk IN.

Этот регистр присутствует только тогда, когда система сконфигурирована для работы в режиме “Device” (включая режимы DRD и OTG). Размер регистра соответствует числу конечных точек IN устройства.

Замечание: поскольку в режиме “Device” используется только один буфер RXFIFO, регистра приоритета DMA для буферов RXFIFO нет.

Поля регистра GTXFIFOPRIDEV описаны в Таблица 26.61.

Таблица 26.61. Глобальный регистр приоритета DMA для буферов FIFO TX устройства

Диапазон	Название	Описание	Исходное значение	Тип доступа
[3:0]	gtxfifopridev	Приоритет буфера TxFIFO устройства (Device TxFIFO priority).	0x0	R/W
[31:4]	reserved_31_n	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0xffff. Ограничение записи: writeAsRead.	0x0	R

26.3.52 Глобальный регистр приоритета DMA для буферов FIFO TX хоста (GTXFIFOPRIHST)

Этот регистр задает относительный уровень приоритета прямого доступа к памяти (Direct Memory Access, DMA) среди буферов TXFIFO хоста (по одному на каждый экземпляр шины) внутри соответствующей скоростной группы (сверхскоростной или высоко/полно/низкоскоростной (SS or HS/FSLS)). Каждый бит [n] регистра управляет приоритетом на доступ (1: высокий, 0: низкий) для каждого TXFIFO[n] внутри скоростной группы. Когда множество буферов TXFIFO конкурируют за работу с DMA в одно и то же время (например, множество TXQ содержат запросы TX DMA и в соответствующих им TXFIFO есть свободное место), арбитр TX DMA предоставляет доступ на основе пакетного обмена по следующим правилам:

1. Среди буферов FIFO в одной скоростной группе (сверхскоростной или высоко/полно/низкоскоростной (SS or HS/FSLS)):

a. буферы TXFIFO, имеющие высокий приоритет, получают доступ по циклическому арбитражу;

b. буферы TXFIFO, имеющие низкий приоритет, получают доступ по циклическому арбитражу только после того, как у буферов TXFIFO с высоким приоритетом больше не остается задач на обработку (либо TXQ опустошены, либо соответствующие им TXFIFO заполнены).

2. Арбитр TX DMA выделяет как приоритетную ту или иную скоростную группу в соответствии с соотношением, запрограммированным в регистре GDMAHLRATIO.

Для пакетов сбора/разброса (scatter-gather), арбитр предоставляет одному и тому же FIFO возможность последовательной отправки запросов к DMA до тех пор, пока не будет завершена работа со всем пакетом.

Этот регистр присутствует только тогда, когда система сконфигурирована для работы в режиме “Host” (включая режимы DRD и OTG). Размер регистра соответствует числу сконфигурированных экземпляров шины USB, например, в конфигурации по умолчанию присутствует 3 экземпляра шины USB (одна сверхскоростная SS, одна высокоскоростная HS, и одна полноскоростная/низкоскоростная FSLS).

Поля регистра GTXFIFOPRIHST описаны в Таблица 26.62.

Таблица 26.62. Глобальный регистр приоритета DMA для буферов FIFO TX хоста

Диапазон	Название	Описание	Исходное значение	Тип доступа
[3:0]	gtxfifoprihst	Приоритет буфера TxFIFO хоста (Host TxFIFO priority).	0x0	R/W
[31:4]	reserved_31_16	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0xffff. Ограничение записи: writeAsRead.	0x0	R

26.3.53 Глобальный регистр приоритета DMA для буферов FIFO RX хоста (GRXFIFOPRIHST)

Этот регистр задает относительный уровень приоритета прямого доступа к памяти (Direct Memory Access, DMA) среди буферов RXFIFO хоста (по одному на каждый экземпляр шины) внутри соответствующей скоростной группы (сверхскоростной или высоко/(полно/низкоскоростной)). Каждый бит [n] регистра управляет приоритетом на доступ (1: высокий, 0: низкий) для каждого RXFIFO[n] внутри скоростной группы. Когда множество буферов RXFIFO конкурируют за работу с DMA в одно и то же время (например, множество RXQ содержат запросы RX DMA и в соответствующих им RXFIFO есть свободное место), арбитр RX DMA предоставляет доступ на основе пакетного обмена по следующим правилам:

1. Среди буферов FIFO в одной скоростной группе (сверхскоростной или высоко/полно/низкоскоростной):

    a. буферы RXFIFO, имеющие высокий приоритет, получают доступ по циклическому арбитражу;

    b. буферы RXFIFO, имеющие низкий приоритет, получают доступ по циклическому арбитражу только после того, как у буферов RXFIFO с высоким приоритетом больше не остается задач на обработку (либо RXQ опустошены, либо соответствующие им RXFIFO заполнены).

2. Арбитр RX DMA выделяет как приоритетную ту или иную скоростную группу в соответствии с соотношением, запрограммированным в регистре GDMAHLRATIO.

Для пакетов сбора/разброса (scatter-gather), арбитр предоставляет одному и тому же FIFO возможность последовательной отправки запросов к DMA до тех пор, пока не будет завершена работа со всем пакетом.

Этот регистр присутствует только тогда, когда система сконфигурирована для работы в режиме “Host” (включая режимы DRD и OTG). Размер регистра соответствует числу сконфигурированных экземпляров шины USB, например, в конфигурации по умолчанию присутствует 3 экземпляра шины USB (одна сверхскоростная, одна высокоскоростная, и одна полноскоростная/низкоскоростная).

Поля регистра GRXFIFOPRIHST описаны в Таблица 26.63.

Таблица 26.63. Глобальный регистр приоритета DMA для буферов FIFO RX хоста

Диапазон	Название	Описание	Исходное значение	Тип доступа
[2:0]	grxfifoprihst	Приоритет буфера RxFIFO хоста (Host RxFIFO priority).	0x0	R/W
[31:3]	reserved_31_16	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0xffff. Ограничение записи: writeAsRead.	0x0	R

26.3.54 Глобальный регистр приоритета DMA для возможностей отладки (GFIFOPRIDBC)

Этот регистр определяет относительный приоритет буферов RXFIFO и TXFIFO, относящихся к режиму DbC (возможностей отладки). Когда режим DbC подключен, этот регистр замещает приоритеты, присвоенные в соответствующих индексах регистров приоритета доступа к DMA для RXFIFO и TXFIFO в режиме “Host”.

Настройки приоритетов заданы на основе низкоприоритетной сверхскоростной (SS) группы скоростей:

1. Стандартный приоритет указывает, что буферы FIFO в режиме DbC считаются идентичными низкоприоритетным сверхскоростным буферам FIFO в режиме “Host”.

2. Низкий приоритет указывает, что буферы FIFO в режиме DbC имеют приоритет ниже, чем все сверхскоростные буферы FIFO в режиме “Host”.

3. Высокий приоритет указывает, что буферы FIFO в режиме DbC имеют приоритет выше, чем низкоприоритетные сверхскоростные буферы FIFO в режиме “Host”, но ниже, чем высокоприоритетные сверхскоростные буферы FIFO в режиме “Host”.

Этот регистр присутствует только тогда, когда система сконфигурирована для работы в режиме возможностей отладки хоста (DbC).

Поля регистра GFIFOPRIDBC описаны в Таблица 26.64.

Таблица 26.64. Глобальный регистр приоритета DMA для возможностей отладки

Диапазон	Название	Описание	Исходное значение	Тип доступа
[1:0]	gfifopridbc	Приоритет на DMA для режима DbC хоста (Host DbC DMA priority).	0x0	R/W
[31:2]	reserved_31_2	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0x3fffffff. Ограничение записи: writeAsRead.	0x0	R

26.3.55 Глобальный регистр соотношения высокого и низкого приоритетов DMA для буферов FIFO в режиме хоста (GDMAHLRATIO)

Этот регистр задает относительный приоритет сверхскоростных (SS) буферов FIFO по отношению к высокоскоростным/(полно/низкоскоростным) (HS/FSLS) буферам FIFO. Арбитр DMA расставляет приоритеты в циклической арбитражной группе скоростных режимов HS/FSLS при каждой выдаче соотношений высоких/низких приоритетов доступа к DMA, так, как это указано в регистре, раздельно для TX и RX.

Для примера, предположим, что все буферы запрашивают доступ одновременно, и соотношение равно 4. Тогда, сначала сверхскоростная группа получает приоритет на 4 пакета, потом высоко/(полно/низкоскоростная) получает приоритет на 1 пакет, потом снова сверхскоростная группа получает приоритет на 4 пакета, высоко/(полно/низкоскоростная) получает приоритет на 1 пакет и так далее.

Если буферы FIFO из обеих скоростных групп не запрашивают доступ одновременно, тогда:

- если сверхскоростная группа получала приоритет 4 раза за последние 4 запроса, тогда высоко/(полно/низкоскоростная) получает приоритет на любой последующий запрос.

- если высоко/(полно/низкоскоростная) группа получала приоритет в последний раз, сверхскоростная группа получает приоритет на следующий запрос.

- если присутствует валидный запрос либо от сверхскоростной, либо от высоко/(полно/низкоскоростной) группы, приоритет выдается всегда, простоя нет.

Этот регистр присутствует, если система сконфигурирована для работы в режиме “Host” (включая режимы DRD и OTG).

Поля регистра GDMAHLRATIO описаны в Таблица 26.65.

Таблица 26.65. Глобальный регистр соотношения высокого и низкого приоритетов DMA для буферов FIFO в режиме хоста

Диапазон	Название	Описание	Исходное значение	Тип доступа
[4:0]	hsttxfifo	Высокий/низкий приоритет буфера TXFIFO хоста на доступ DMA (Host TXFIFO DMA High-Low Priority).	0x8	R/W
[7:5]	reserved_7_5	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0x7. Ограничение записи: writeAsRead.	0x0	R
[12:8]	hstrxfifo	Высокий/низкий приоритет буфера RXFIFO хоста на доступ DMA (Host RXFIFO DMA High-Low Priority).	0x8	R/W
[31:13]	reserved_31_13	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0x7ffff. Ограничение записи: writeAsRead.	0x0	R

26.3.56 Глобальный регистр настройки длины фреймов (GFLADJ)

Этот регистр предоставляет программному обеспечению возможность управлять поведением системы по отношению к таймеру начала фрейма SOF (Start of Frame) и таймеру пакета изохронной временной метки ITP (Isochronous Timestamp Packet), а также функционалом таймера фреймов. Он предоставляет возможность переопределить сигнал боковой полосы частот fladj_30mhz_reg (sideband signal). К тому же, он подключает возможность запускать счетчики таймеров фреймов SOF или ITP исключительно через синхросигнал ref_clk. Управление счетчиками SOF или ITP через ref_clk облегчает аппаратное управление питанием каналов (LPM) в режиме “Host”.

Поля регистра GFLADJ описаны в Таблица 26.66.

Таблица 26.66. Глобальный регистр настройки длины фреймов

Диапазон	Название	Описание	Исходное значение	Тип доступа
[5:0]	GFLADJ_30MHZ	Это поле указывает значение, которое используется для того, чтобы настраивать длину фрейма вместо того, чтобы оценивать её по входному сигналу боковых частот fladj_30mhz_reg. Оно подключает настройку длины фрейма после изготовления микросхемы (post-silicon) на тот случай, если входной сигнал fladj_30mhz_reg подсоединен к неверному значению или невалиден. Подробнее процесс установки значения этого поля описан в Разделе 5.2.4, «Регистр настройки длины фрейма (Frame Length Adjustment Register, FLADJ)» спецификации xHCI.	0x0	R/W
[6]	reserved_6	Зарезервировано для дальнейшего использования. Контролепригодность: writeAsRead. Маска сброса: 0x1. Ограничение записи: writeAsRead.	0x0	R
[7]	GFLADJ_30MHZ_SDBND_SEL	Это поле указывает, что следует использовать при настройке длины фрейма для SOF/ITP: входной сигнал fladj_30mhz_reg или GFLADJ.GFLADJ_30MHZ. Когда этот бит имеет значение: 1: контроллер использует значение поля регистра GFLADJ.GFLADJ_30MHZ; 0: контроллер использует значение входного сигнала fladj_30mhz_reg.	0x0	R/W
[21:8]	GFLADJ_REFCLK_FLADJ	Это поле указывает, какие настройки длины фрейма следует применять, когда счетчик SOF/ITP управляется синхросигналом ref_clk. Значение этого поля используется для настройки: 1) интервала ITP, когда в GCTL[SOFITPSYNC] установлено значение «1»; 2) интервалов SOF и ITP, когда в GLADJ.GFLADJ_REFCLK_LPM_SEL установлено значение «1». В это поле следует программировать ненулевую величину только в том случае, если в GFLADJ_REFCLK_LPM_SEL или GCTL.SOFITPSYNC установлено значение «1». Значение вычисляется следующим образом:	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		FLADJ_REF_CLK_FLADJ = ((125000/ref_clk_period_integer) - (125000/ref_clk_period)) * ref_clk_period, где: - ref_clk_period_integer - целое значение периода ref_clk, полученного за счёт округления десятичного (дробного) значения, которое запрограммировано в поле GUCTL.REF_CLK_PERIOD. - ref_clk_period - период ref_clk, включая дробную величину. Пример: если ref_clk = 24 МГц, тогда: GUCTL.REF_CLK_PERIOD = 41; GFLADJ.GLADJ_REFCLK_FLADJ = ((125000/41) - (125000/41.6666))*41.6666 = 2032 (отбрасывая дробную часть). Если ref_clk = 48 МГц, тогда: GUCTL.REF_CLK_PERIOD = 20; GFLADJ.GLADJ_REFCLK_FLADJ = ((125000/20) - (125000/20.8333))*20.8333 = 5208 (отбрасывая дробную часть).		
[22]	reserved_22	Зарезервировано для дальнейшего использования. Контролепригодность: writeAsRead. Маска сброса: 0x1. Ограничение записи: writeAsRead.	0x0	R
[23]	GFLADJ_REFCLK_LPM_SEL	Этот бит подключает функционал управления счетчиками SOF/ITP через ref_clk. В этот бит не следует устанавливать «1», если в GCTL.SOFITPSYNC установлено «1». Аналогичным образом, если в GFLADJ_REFCLK_LPM_SEL установлено «1», в GCTL.SOFITPSYNC не следует устанавливать «1». Когда в GFLADJ_REFCLK_LPM_SEL установлено «1», отключается перехват управления приостановкой первого порта USB 2.0 PHY (UTMI/ULPI) состояниями портов USB 3.0. Например, при работе с PHY от Synopsys, сигнал COMMONONN может быть привязан к «1». Следует отметить, что поддерживаемые в этом режиме частоты ref_clk : 16/17/19.2/20/24/39.7/40 МГц. Системный сигнал utmi_clk[0] должен быть соединен с сигналом FREECLK от PHY. Замечание: если в этот бит устанавливается значение «1», в бит GUSB2PHYCFG.U2_FREECLK_EXISTS необходимо установить «0».	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
[30:24]	GFLADJ_REFCLK_240MHZ_DECR	Это поле указывает значение декремента, который контроллер использует для каждого сигнала ref_clk для того, чтобы вычислить таймер фрейма в единицах синхросигнала на частоте 240 МГц. В это поле следует программировать ненулевое значение только если в GFLADJ_REFCLK_LPM_SEL установлено значение «1». Значение поля вычисляется следующим образом: GFLADJ_REFCLK_240MHZ_DECR = 240/ref_clk_frequency. Примеры: если ref_clk = 24 МГц, тогда: GUCTL.REF_CLK_PERIOD = 41; GFLADJ.GFLADJ_REFCLK_240MHZ_DECR = 240/24 = 10. если ref_clk = 48 МГц, тогда: GUCTL.REF_CLK_PERIOD = 20; GFLADJ.GFLADJ_REFCLK_240MHZ_DECR = 240/48 = 5. если ref_clk = 17 МГц, тогда: GUCTL.REF_CLK_PERIOD = 58; GFLADJ.GFLADJ_REFCLK_240MHZ_DECR = 240/17 = 14.	0x0	R/W
[31]	GFLADJ_REFCLK_240MHZDECR_PLS1	Это поле указывает, что что значение декремента, который контроллер применяет к каждому ref_clk должно быть равным GFLADJ_REFCLK_240MHZ_DECR и GFLADJ_REFCLK_240MHZ_DECR+1 поочередно для каждого ref_clk. Следует устанавливать значение «1» в этот бит только в том случае, если в GFLADJ_REFCLK_LPM_SEL установлено «1» и дробная часть значения 240/ref_frequency больше или равна 0,5. Примеры. Если ref_clk = 19.2 МГц, тогда: - GUCTL.REF_CLK_PERIOD = 52; - GFLADJ.GFLADJ_REFCLK_240MHZ_DECR = (240/19.2) = 12.5; - GFLADJ.GFLADJ_REFCLK_240MHZDECR_PLS1 = 1. Если ref_clk = 24 МГц, тогда: - GUCTL.REF_CLK_PERIOD = 41;	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		- GFLADJ.GFLADJ_REFCLK_240MHZ_DECR = (240/24) = 10; - GFLADJ.GFLADJ_REFCLK_240MHZDECR_PLS1 = 0.		

26.4 Блок регистров устройства USB3

Карта регистров DWC_usb3_block_dev приведена в Таблица 26.67.

Таблица 26.67. Карта регистров DWC_usb3_block_dev

№	Регистр	Описание	Смещение
1	DCFG	Регистр конфигурирования устройства (Device Configuration Register)	0xc700
2	DCTL	Регистр управления устройством (Device Control Register)	0xc704
3	DEVTEN	Регистр подключения событий устройства (Device Event Enable Register)	0xc708
4	DSTS	Регистр статуса устройства (Device Status Register)	0xc70C
5	DGCMDPAR	Регистр параметра универсальной команды устройства (Device Generic Command Parameter Register)	0xc710
6	DGCMD	Регистр универсальной команды устройства (Device Generic Command Register)	0xc714
7	DALEPENA	Регистр подключения активных конечных точек USB устройства (Device Active USB Endpoint Enable Register)	0xc720
8	Rsvd[0:31]	Зарезервировано.	0xc724
9	DEPCMDPAR2[0:7]	Регистр параметра 2 команды физической конечной точки-n устройства (Device Physical Endpoint-n Command Parameter 2 Register (DEPCMDPAR2n))	0xc800
10	DEPCMDPAR1[0:7]	Регистр параметра 1 команды физической конечной точки-n устройства (Device Physical Endpoint-n Command Parameter 1 Register (DEPCMDPAR1n))	0xc804
11	DEPCMDPAR0[0:7]	Регистр параметра 0 команды физической конечной точки-n устройства (Device Physical Endpoint-n Command Parameter 0 Register (DEPCMDPAR0n))	0xc808
12	DEPCMD[0:7]	Регистр команд физической конечной точки-n устройства (Device Physical Endpoint-n Command Register)	0xc80c
13	DEV_IMOD[0]	Регистр модерации прерываний устройства (Device Interrupt Moderation Register (DEV_IMOD))	0xca00

26.4.1 Регистр конфигурирования устройства (DCFG)

Этот регистр конфигурирует систему в режиме “Device” после включения питания или после определенных команд управления или перебора (enumeration).

Не следует вносить изменения в этот регистр после программирования начальных параметров.

Поля регистра DCFG описаны в Таблица 26.68.

Таблица 26.68. Регистр конфигурирования устройства

Диапазон	Название	Описание	Исходное значение	Тип доступа
[2:0]	DEVSPD	Скорость устройства (Device Speed). Указывает скорость соединения, которую приложение требует от системы, или максимальную скорость, которую приложение может поддерживать. Однако, фактическая скорость шины будет определена только после завершения последовательности чирпирования (chirp sequence) и основывается на скорости хоста USB, к которому подключена система. 3'b100: сверхскоростной режим (такт PHY USB 3.0 - 125 МГц или 250 МГц); 3'b000: высокоскоростной режим (такт PHY USB 2.0 - 30 МГц или 60 МГц); 3'b001: полноскоростной режим (такт PHY USB 2.0 - 30 МГц или 60 МГц); Значения: - 0x4 (сверхскоростной): сверхскоростной режим (такт PHY USB 3.0 - 125 МГц или 250 МГц); - 0x0 (высокоскоростной): высокоскоростной режим (такт PHY USB 2.0 - 30 МГц или 60 МГц); - 0x1 (полноскоростной): полноскоростной режим (такт PHY USB 2.0 - 30 МГц или 60 МГц); Контролепригодность: без ограничений.	0x4	R/W
[9:3]	DEVADDR	Адрес устройства (Device Address). Приложение должно исполнять следующие действия: - программировать это поле после каждого запроса SetAddress; - сбрасывать это поле до 0 после сброса USB.	0x0	R/W
[11:10]	reserved_10_11	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0x3. Ограничение записи: writeAsRead.	0x2	R
[16:12]	INTRNUM	Номер прерывания (Interrupt number).	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Указывает номер прерывания/EventQ, на котором генерируются относящиеся к устройству, неспецифические к конечной точке прерывания (см. DEVT).		
[21:17]	NUMP	Число буферов приёма (Number of Receive Buffers). Указывает число буферов приёма, отчёт о котором пишется в пакете ACK TP (“acknowledge” transmission packet). Контроллер DWC_usb3 использует это поле для конечных точек, не относящихся к управляющим (non-control), если в GRXTHRCFG.UsbRxPktCntSel установлен «0». Приложение может программировать это значение на основе размера RxFIFO, размеров буферов, запрограммированных в дескрипторах, и системного запаздывания. Для конечной точки типа OUT, это поле управляет числом буферов приема, отчет о которых пишется в поле NumP пакета ACK TP, переданного системой. Замечание: когда подключена функция отладки (Debug Capability), этот бит используетя в режиме “Host”.	0x4	R/W
[22]	LPMCAP	Режим LPM (Link Power Management, управление питанием канала) возможен (LPM Capable). Приложение использует этот бит для управления возможностями режима LPM системы DWC_usb3. Если система работает как устройство, не обладающее возможностями LPM, оно не может отвечать на транзакции LPM. 1'b0: возможности LPM не подключены; 1'b1: возможности LPM подключены.	0x0	R/W
[23]	IgnStrmPP	Игнорирование бита PP и потока (IgnoreStreamPP). Этот бит влияет только на конечные точки типа bulk, способные к потоковой передаче (stream-capable). Когда в этот бит установлен «0» и контроллер получает пакет данных, в котором в бит «пакета в ожидании» (Packet Pending, PP) установлен 0 для конечных точек OUT; или он получает пакет ACK, в котором в поле NumP установлен 0 и в PP установлен 0 для конечных точек IN; система предпринимает попытку поиска другого потока для инициации к хосту (CStream). Однако, существуют две ситуации, в которых такое поведение неоптимально: - когда хост устанавливает PP=0 даже несмотря на то, что он не завершил поток;	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		- когда конечная точка устройства сконфигурирована с одним ресурсом передачи (one transfer resource) и, таким образом, не имеет никаких других потоков для инициации к хосту. Когда в этот бит установлена «1», система игнорирует бит «пакета в ожидании» в целях выбора потока и не ищет другого потока, когда получает DP(PP=0) или ACK(NumP=0, PP=0). Это может улучшить производительность, когда пропускная способность системной шины устройства низкая или хост отвечает на системные передачи ERDY очень быстро.		
[24]		Зарезервировано.	0	
[31:25]	reserved_31_25	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0x3f. Ограничение записи: writeAsRead.	0x0	R

26.4.2 Регистр управления устройством (DCTL)

Когда гибернация не подключена через поле GCTL.GblHibernationEn:

- в поля CSS, CRS, L1HibernationEn, и KeepConnect можно писать любое значение;
- поля L1HibernationEn и KeepConnect всегда возвращают 0 при чтении из них в таком состянии отключенной гибернации.

Поля регистра DCTL описаны в Таблица 26.69.

Таблица 26.69. Регистр управления устройством

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	reserved_0	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0x1. Ограничение записи: writeAsRead.	0x0	R
[4:1]	TSTCTL	Управление тестами. 4'b000: режим тестирования отключен; 4'b001: режим Test_J; 4'b010: режим Test_K; 4'b011: режим Test_SE0_NAK; 4'b100: режим Test_Packet; 4'b101: режим Test_Force_Enable; Другие: зарезервировано. Контролепригодность: writeAsRead.	0x0	R/W
[8:5]	ULSTCHNGREQ	Программное обеспечение пишет в это поле, чтобы инициировать запрос об изменении состояния USB/Link.	0x0	W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Изменение этого поля указывает новый запрос к системе. Если программному обеспечению нужно инициировать тот же самый запрос сразу за предыдущим, необходимо записать 0 в это поле между этими двумя запросами. Результат запроса о смене состояния отражается в состоянии USB/Link в DSTS. Эти биты самоочищаются в приостановленном состоянии выхода из уровня MAC (подуровня управления доступом к среде, medium access control) (MAC Layer exiting suspended state). Если программное обеспечение обновляет остальные поля регистра DCTL и не планируется запускать никаких изменений состояния канала, тогда необходимо писать 0 в это поле. Партнер по удаленному каналу, как правило, входит в режим “SS Compliance” (соответствия сверхскоростному режиму) и контролирует его. Подробнее это описано в спецификации USB 3.0. В противном случае, локальный канал можно напрямую ввести в режим соответствия путем сброса сверхскоростного канала, установив 0 в бит RUN/STOP. Если затем записать «10» в поле смены состояния USB/Link и «1» в RUN/STOP, канал войдет в режим соответствия. После попадания в режим соответствия можно попеременно писать «0» и «10» в это поле для совершенствования структуры соответствия. В сверхскоростном режиме (SS): Значение для переключения/действия запрошенного состояния канала: 0 : никакого действия; 4 : SS.Disabled (сверхскоростной режим отключен); 5 : Rx.Detect (обнаружение); 6 : SS.Inactive (неактивен); 8 : возобновление; 10 : соответствие; Другие значения: зарезервировано. В режиме HS/FS/LS (высоко/полно/низкоскоростном): Значение для переключения запрошенного состояния USB: 8 : запрос удаленного пробуждения (Remote wakeup request); Другие значения: зарезервировано. Запрос удаленного пробуждения должен быть отправлен через 2 мкс после перехода устройства в состояние приостановки (значение DSTS[21:18] - 3, следует обращаться к таблице «поля регистра DSTS» в справочнике).		

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Замечание: после выхода из гибернации, программное обеспечение обязано писать 8 (возобновление) в это поле для подтверждения выхода из состояния приостановки. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.		
[9]	ACCEPTU1ENA	Принятие подключения U1. 1'b0: система отвергает U1, кроме случая, когда установлен бит Force_LinkPM_Accept (значение по умолчанию); 1'b1: система принимает переключение в состояние U1, если со стороны приложения ничто не ожидает обработки. При сбросе USB, этот бит обнуляется аппаратно. Программное обеспечение устанавливает этот бит после получения команды установки конфигурации SetConfiguration.	0x0	R/W
[10]	INITU1ENA	Инициация подключения U1. 1'b0: не может инициировать U1 (по умолчанию); 1'b1: может инициировать U1. При сбросе USB, этот бит обнуляется аппаратно. Программное обеспечение устанавливает этот бит после получения SetFeature(U1_ENABLE), и очищает этот бит, когда получено ClearFeature(U1_ENABLE). Если DCTL[9] (AcceptU1Ena) имеет значение 0, канал немедленно выходит из состояния U1.	0x0	R/W
[11]	ACCEPTU2ENA	Принятие подключения U2. 1'b0: система отвергает U2, кроме случая, когда установлен бит Force_LinkPM_Accept (значение по умолчанию); 1'b1: система принимает переключение в состояние U2, если со стороны приложения ничто не ожидает обработки. При сбросе USB, этот бит обнуляется аппаратно. Программное обеспечение устанавливает этот бит после получения команды установки конфигурации SetConfiguration.	0x0	R/W
[12]	INITU2ENA	Инициация подключения U2. 1'b0: не может инициировать U2 (по умолчанию); 1'b1: может инициировать U2. При сбросе USB, этот бит обнуляется аппаратно. Программное обеспечение устанавливает этот бит после получения SetFeature(U2_ENABLE), и очищает этот бит когда получено ClearFeature(U2_ENABLE). Если DCTL[11] (AcceptU2Ena) имеет значение 0, канал немедленно выходит из состояния U2.	0x0	R/W
[15:13]	reserved_15_13	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0x7.	0x0	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Ограничение записи: writeAsRead.		
[16]	CSS	Сохранение состояния контроллера (Controller Save State, CSS). Эта команда аналогична биту USBCMD.CSS в режиме “Host”. Она инициирует процесс сохранения. Когда программное обеспечение устанавливает «1» в этот бит, контроллер немедленно устанавливает «1» в DSTS.SSS. Когда контроллер завершил процесс сохранения, он устанавливает «0» в DSTS.SSS. Замечание: при чтении это поле всегда возвращает «0». Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W
[17]	CRS	Восстановление состояния контроллера (Controller Restore State, CRS). Эта команда аналогична биту USBCMD.CRS в режиме “Host”. Она инициирует процесс восстановления. Когда программное обеспечение устанавливает «1» в этот бит, контроллер немедленно устанавливает «1» в DSTS.RSS. Когда контроллер завершил процесс восстановления, он устанавливает «0» в DSTS.RSS. Замечание: при чтении это поле всегда возвращает «0». Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R/W
[18]	L1HibernationEn	Когда этот бит устанавливается вместе с KeepConnect, система устройства генерирует событие запроса гибернации (Hibernation Request Event) если L1 подключено, и значение HIRD в токене LPM выше, чем граничное значение, запрограммированное в DCTL.HIRD_Thres. Система не выходит из состояния LPM L1 до тех пор, пока программное обеспечение не запишет “Recovery” (восстановление) в поле DCTL.ULStChngReq. Это предотвращает тупиковые ситуации, когда устройство входит в режим гибернации и одновременно хост пытается выйти из L1. Замечание: если гибернация отключена, то есть, GCTL[1].GblHibernationEn = 0, этот бит привязан к нулю. Изменчивость: true.	0x0	R/W
[19]	KeepConnect	Удерживать соединение (Keep Connect). Когда установлена «1», этот бит подключает модель программирования с сохранением и восстановлением, не позволяя системе	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		отсоединяться от хоста, когда в DCTL.RunStop установлен «0». Он также подключает генерацию события запроса гибернации (Hibernation Request Event) когда канал переходит в состояние U3 или L2. Система устройства отсоединяется от хоста, когда в DCTL.RunStop установлен «0». Этот бит указывает, придерживаться ли далее этого поведения («0»), или постановить, что система не должна отсоединяться, когда в RunStop установлен 0 («1»). Этот бит также предотвращает автоматический переход LTSSM в U0/L0 когда запросы хоста возобновляются из U3/L2. Замечание: если гибернация отключена, то есть, GCTL[1].GblHibernationEn = 0, этот бит привязан к нулю. Изменчивость: true.		
[23:20]	LPM_NYET_thres	Граничное значение NYET в LPM (LPM NYET Threshold). Когда подключен корректировочный лист LPM: биты [23:20]: граничное значение ответа NYET в LPM (LPM_NYET_thres). Ответ о согласовании (handshake response) токену LPM, заданный приложением устройства. Ответ зависит от DCFG.LPMCap. - Если DCFG.LPMCap имеет значение 1'b0 - система всегда отвечает таймаутом (таким образом, ответа нет); - если DCFG.LPMCap имеет значение 1'b1 - система отвечает пакетом ACK на успешную транзакцию LPM, что требует удовлетворения всех следующих условий: 1) ошибки PID или CRC5 как в токене EXT, так и в токене LPM отсутствуют (если это неверно, неактивность приводить к ошибке таймаута); 2) в буфере TxFIFO нет данных, ожидающих очереди, а буфер RxFIFO пуст (в противном случае отправляется NYET); 3) значение BESL (Best Effort Service latency, запаздывание наилучшей попытки обслуживания) в токене LPM меньше или равно LPM_NYET_thres[3:0].	0xf	R/W
[28:24]	HIRDTHRES	Граничное значение HIRD (host-initiated resume duration, длительность возобновления, инициированного хостом) (HIRD_Thres). Система устанавливает сигналы вывода utmi_l1_suspend_n и utmi_sleep_n (см. таблицу «Сигналы интерфейса LPM» в справочнике) на основе данного сигнала. 1) Система устанавливает utmi_l1_suspend_n, чтобы поместить PHY в глубокий режим низкого питания в L1, когда выполняются оба следующих условия:	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		- значение HIRD больше или равно значению в DCTL.HIRD_Thres[3:0]; - в HIRD_Thres[4] установлено 1'b1. 2) Система устанавливает utmi_sleep_n на L1, когда выполняется одно из следующих условий: - если значение HIRD меньше чем HIRD_Thres[3:0] или - если в HIRD_Thres[4] установлено 1'b0. Замечание: во время сверхскоростного режима работы в это поле должен быть установлен «0».		
[29]	reserved_29	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0x1. Ограничение записи: writeAsRead.	0x0	R
[30]	CSFTRST	Мягкий сброс системы (Core Soft Reset). Сбрасывает все домены синхросигналов следующим образом: - этот бит очищает прерывания и все CSRs кроме регистров GSTS, GSNPSID, GGPIO, GUID, групп регистров GUSB2PHYCFGn, групп регистров GUSB3PIPECTLn, DCFG, DCTL, DEVTEN, и DSTS; - все конечные автоматы модуля (кроме блока подичненной шины SoC) сбрасываются к состоянию IDLE (простоя), и все буферы TxFIFOs и RxFIFO очищаются; - любые транзакции на главную шину SoC прерываются как можно скорее, после корректного завершения последней фазы передачи данных на шине SoC. Любые транзакции на USB прерываются незамедлительно. Приложение может в любой момент написать в этот бит, чтобы сбросить систему. Это самоочищающийся бит, система очищает этот бит после того, как вся необходимая логика сбрасывается в системе, что может потребовать нескольких тактов, в зависимости от текущего состояния системы. После того, как бит очищен, программное обеспечение обязано ожидать как минимум 3 такта синхросигнала PHY перед обращением к домену PHY (задержка синхронизации). Как правило, в процессе разработки программного обеспечения и также при динамическом изменении битов выбора PHY в регистрах конфигурации USB, перечисленных выше, применяется программный сброс. Когда изменения вносятся в PHY, в домене PHY выбирается и используется соответствующий синхросигнал для PHY. После того, как выбран новый синхросигнал, домен PHY должен быть сброшен для корректной работы. Замечание: программирование случайных данных в это поле провоцирует побочный	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		эффект. Не рекомендуется применять тест Bit Bash на этом регистре. Контролепригодность: writeAsRead. Изменчивость: true.		
[31]	RUN_STOP	Пуск/Стоп (Run/Stop). Программное обеспечение пишет в этот бит «1» для запуска работы с контроллером устройства. Для прекращения работы с контроллером устройства, программное обеспечение обязано убрать любые активные передачи и записать 0 в этот бит. Когда контроллер остановлен, он устанавливает бит DSTS.DevCtrlHlt, когда система находится в состоянии простоя и нижний уровень завершает процесс отсоединения. Бит Run/Stop должен быть использован в следующих случаях нижеописанным образом: - после сброса при включении питания и инициализации CSR, необходимо, чтобы программное обеспечение записало «1» в этот бит, чтобы запустить контроллер устройства. Контроллер не сигналит о подсоединении к хосту до тех пор, пока этот бит не установлен. - программное обеспечение использует этот бит для управления контроллером устройства, чтобы произвести мягкое отсоединение. Когда программное обеспечение пишет 0 в этот бит, хост не видит, что устройство подключено. Контроллер устройства остается в отсоединенном состоянии до тех пор, пока программное обеспечение не запишет 1 в этот бит. Минимальная длительность удержания этого бита очищенным задана в замечании ниже. Если программное обеспечение пытается подсоединиться после мягкого отсоединения, или обнаруживает событие отсоединения, оно обязано установить 5 в DCTL[8:5] перед переустановкой бита Run/Stop. - когда USB или канал находятся в состоянии пониженного питания и выбрана конфигурация “Two Power Rails” (двух шин питания), программное обеспечение пишет «0» в этот бит, чтобы указать, что оно собирается отключить системную шину питания (Core Power Rail). После того, как программное обеспечение снова включает системную шину питания и заново инициализирует контроллер устройства, оно обязано установить этот бит для запуска контроллера устройства. Подробнее описано в Разделе «Работа при низком питании» справочника. Замечание: ниже перечислены минимальные времена в различных условиях, на протяжении которых бит мягкого отсоединения (SftDiscon)	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		должен быть установлен, чтобы хост USB определил отсоединение устройства. 30 мс: - для сверхскоростного режима (SuperSpeed), когда устройство находится в состояниях: приостановки (Suspended), простоя (Idle), передачи(Transmit), или получения(Receive). 10 мс: - для высокоскоростного режима, когда устройство находится в состояниях: приостановки (Suspended), простоя (Idle), или вне приостановки/простоя (при выполнении транзакций); - для полно-/низкоскоростных режимов, когда устройство находится в состояниях: приостановки (Suspended), простоя (Idle), или вне приостановки/простоя (при выполнении транзакций). Для компенсации флуктуаций синхросигнала рекомендуется, чтобы приложение прибавляло добавочную задержку к указанной минимальной длительности.		

26.4.3 Регистр подключения событий устройства (DEVTEN)

Этот регистр управляет генерацией специфичных для устройства событий (см. Раздел “Event Buffer Content for Device-Specific Events (DEVT)” (Содержимое буфера событий для специфичных для устройства событий (DEVT)) в справочнике). Если в бит подключения установлен 0, событие не будет генерироваться.

Поля регистра DEVTEN описаны в Таблица 26.70.

Таблица 26.70. Регистр подключения событий устройства

Диапазон	Название	Описание	Исходное значение	Тип доступа
[0]	DISSCONNEVTEN	Подключение события обнаружения отсоединения (Disconnect Detected Event Enable).	0x0	R/W
[1]	USBRSTEVTEN	Подключение события сброса USB (USB Reset Enable).	0x0	R/W
[2]	CONNECTDONEEVTEN	Подключение события об установке соединения (Connection Done Enable).	0x0	R/W
[3]	ULSTCNGEN	Подключение события смены состояния канала/USB (USB/Link State Change Event Enable).	0x0	R/W
[4]	WKUPEVTEN	Подключение события обнаружения возобновления U3/L2 или U3/L2L1 (U3/L2 or U3/L2L1 Resume Detected Event Enable). Замечание: - если GUCTL1[DEV_DECOUPLE_L1L2_EVT] подключен, тогда этот бит предназначен для подключения события обнаружения возобновления U3/L2.	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		- если GUCTL1[DEV_DECOUPLE_L1L2_EVT] не подключен, тогда этот бит предназначен для подключения события обнаружения возобновления U3/L2L1.		
[5]	HibernationReqEvtEn	Этот бит подключает/отключает генерацию события запроса о гибернации (Hibernation Request Event).	0x0	R/W
[6]	U3L2L1SuspEn	Подключение события приостановки U3/L2 или U3/L2L1 (U3/L2 or U3/L2L1 Suspend Event Enable). Замечание: - если GUCTL1[DEV_DECOUPLE_L1L2_EVT] подключен, тогда этот бит предназначен для подключения события приостановки U3/L2. - если GUCTL1[DEV_DECOUPLE_L1L2_EVT] не подключен, тогда этот бит предназначен для подключения события приостановки U3/L2L1.	0x0	R/W
[7]	SOFTEVTEN	Запуск (u)frame (Start of (u)frame).	0x0	R/W
[8]	L1SUSPEN	Подключение события приостановки L1 (L1 Suspend Event Enable). Замечание: этот бит относится к подключению события приостановки L1 только если GUCTL1[DEV_DECOUPLE_L1L2_EVT] подключен.	0x0	R/W
[9]	ERRTICERREVTEN	Подключение события нерегулярной ошибки (Erratic Error Event Enable).	0x0	R/W
[10]	reserved_10	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0x1. Ограничение записи: writeAsRead.	0x0	R
[11]	reserved_11	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0x1. Ограничение записи: writeAsRead.	0x0	R
[12]	VENDEVTSTRCVDEN	Полученное событие о тесте протокола управления каналом (LMP, link management protocol) устройства от поставщика (Vendor Device Test LMP Received Event).	0x0	R/W
[13]		Зарезервировано.	0	
[14]	L1WKUPEVTEN	Подключение события обнаружения возобновления L1 (L1 Resume Detected Event Enable). Замечание: если подключено событие GUCTL1[DEV_DECOUPLE_L1L2_EVT], тогда этот бит предназначен для подключения события обнаружения возобновления L1.	0x0	R/W
[15]	reserved_15	Зарезервировано. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.	0x0	R
[16]	ECCERREN	Подключение ошибки ECC (код, корректирующий ошибки, error-correcting code) (ECC Error Enable).	0x0	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Если в этот бит установлена 1, контроллер отправляет программному обеспечению отчёт об ошибке ECC, когда внутри возникает некорректируемая ошибка ECC. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.		
[31:17]	reserved_31_17	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0x7fff. Ограничение записи: writeAsRead.	0x0	R

26.4.4 Регистр статуса устройства (DSTS)

Этот регистр указывает статус контроллера устройства по отношению к событиям, связанным с USB.

Поля регистра DSTS описаны в Таблица 26.71.

Таблица 26.71. Регистр статуса устройства

Диапазон	Название	Описание	Исходное значение	Тип доступа
[2:0]	CONNECTSPD	Скорость соединения (Connected Speed). Указывает режим скорости, с которым система DWC_usb3 восстанавливается после определения скорости через последовательность чирпирования. 3'b100 – сверхскоростной режим (синхросигнал PHY работает на 125 или 250 МГц); 3'b000 – высокоскоростной режим (синхросигнал PHY работает на 30 или 60 МГц); 3'b001 – полноскоростной режим (синхросигнал PHY работает на 30 или 60 МГц). Низкоскоростной режим не поддерживается для устройств, использующих PHY UTMI+. Значения: – 0x4 (SuperSpeed): сверхскоростной режим (синхросигнал PHY работает на 125 или 250 МГц); – 0x0 (HighSpeed): высокоскоростной режим (синхросигнал PHY работает на 30 или 60 МГц); – 0x1 (FullSpeed): полноскоростной режим (синхросигнал PHY работает на 30 или 60 МГц). Контролепригодность: writeAsRead. Изменчивость: true. Ограничения записи: writeAsRead.	0x4	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
[16:3]	SOFFN	Номер фрейма/микрофрейма полученного начала фрейма (SOF) (Frame/Microframe Number of the Received SOF). Когда система работает в сверхскоростном режиме: – [16:3] указывает номер uframe/ITP. Когда система работает в высокоскоростном режиме: – [16:6] указывает номер фрейма; – [5:3] указывает номер микрофрейма. Когда система работает в полноскоростном режиме: – [16:4] не используется. Программное обеспечение может игнорировать эти 3 бита; – [13:3] указывает номер фрейма. Контролепригодность: writeAsRead. Изменчивость: true. Ограничение записи: writeAsRead.	0x0	R
[17]	RXFIFOEMPTY	Буфер RxFIFO пуст (RxFIFO Empty). Контролепригодность: writeAsRead. Изменчивость: true. Ограничение записи: writeAsRead.	0x1	R
[21:18]	USBLNKST	USBLNKST. Состояние USB/канала (USB/Link State). В сверхскоростном режиме (SS): состояние LTSSM 4'h0: U0; 4'h1: U1; 4'h2: U2; 4'h3: U3; 4'h4: SS_DIS; 4'h5: RX_DET; 4'h6: SS_INACT; 4'h7: POLL; 4'h8: RECOV; 4'h9: HRESET; 4'ha: CMPLY; 4'hb: LPBK; 4'hf: возобновление/сброс. В высоко/полно/низкоскоростном режиме (HS/FS/LS): 4'h0: включено; 4'h2: сон (L1); 4'h3: приостановка (L2); 4'h4: отсоединение (по умолчанию); 4'h5: ранняя приостановка (валидно только когда гибернация отключена, GCTL[1].GblHibernationEn = 0); 4'he: сброс (валидно только когда гибернация подключена, GCTL[1].GblHibernationEn = 1);	0x4	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
		4'hf: возобновление (валидно только когда гибернация подключена, GCTL[1].GblHibernationEn = 1). Состояние канала «возобновление/сброс» указывает, что система получила запрос на возобновление или запрос на сброс USB от хоста в то время, пока канал находился в гибернации. Программное обеспечение должно записать «8» (возобновление) в поле DCTL.ULStChngReq, чтобы засвидетельствовать запрос возобновления/сброса. Когда гибернация подключена, GCTL[1].GblHibernationEn = 1, это поле USBLnkSt валидно только когда в DCTL[31].Run/Stop установлена «1» и DSTS[29].DCNRD = 0. Замечание: если SSIC подключен во время выхода из состояния низкого питания, поле USBLnkSt указывает возобновление/сброс даже для состояния отсоединения, поскольку возобновление предшествует отсоединению. Контролепригодность: writeAsRead. Изменчивость: true. Ограничение записи: writeAsRead.		
[22]	DEVCTRLHLT	Контроллер устройства приостановлен (Device Controller Halted). Когда в бит Run/Stop регистра DCTL устанавливают 1, в этот бит устанавливают 0. Система устанавливает 1 в этот бит, когда, после того как SW установит в Run/Stop 0, система находится в состоянии простоя и нижний слой завершает процесс отсоединения. Когда Halted=1, система не генерирует события устройства. Замечание: – система не устанавливает 1 в этот бит, если GEVNTCOUNTn имеет некое валидное значение. Программному обеспечению необходимо засвидетельствовать события, которые генерируются (путем записи в GEVNTCOUNTn) в то время, как оно ожидает, пока в этот бит будет установлена 1. – когда подключена модерация прерываний, может быть задержка в возбуждении линии прерывания, когда счетчик событий не является нулевым. Программное обеспечение должно читать регистры GEVNTCOUNT напрямую и опознавать их. Контролепригодность: writeAsRead. Изменчивость: true. Ограничение записи: writeAsRead.	0x1	R
[23]	COREIDLE	Простой системы (Core Idle).	0x1	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Этот бит указывает, что система завершила передачу всех данных буфера RxFIFO в системную память, перезаписав все завершенные дескрипторы и выставив 0 во все счётчики событий. Замечание: во время тестирования на значения сброса, следует маскировать (mask out) значение чтения. Этот бит отражает изменяющееся состояние системы и значение в нём – не статическое. Контролепригодность: нетестируемо. Изменчивость: true. Ограничение записи: writeAsRead.		
[24]	SSS	Статус состояния сохранения (Save State Status). Этот бит аналогичен USBSTS.SSS в режиме “Host”. Когда контроллер только что завершил процесс сохранения, он завершает команду установкой 0 в DSTS.SSS. Контролепригодность: writeAsRead. Изменчивость: true. Ограничение записи: writeAsRead.	0x0	R
[25]	RSS	Статус состояния восстановления (Restore State Status). Этот бит аналогичен USBSTS.RSS в режиме “Host”. Когда контроллер только что завершил процесс восстановления, он завершает команду установкой 0 в DSTS.RSS. Контролепригодность: writeAsRead. Изменчивость: true. Ограничение записи: writeAsRead.	0x0	R
[27:26]	reserved_27_26	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0x3. Изменчивость: true. Ограничение записи: writeAsRead.	0x0	R
[28]	SRE	Ошибка восстановления сохранения (Save Restore Error). Не поддерживается в настоящее время. Маска сброса: 0x1. Изменчивость: true.	0x0	R/W
[29]	DCNRD	Контроллер устройства не готов к работе (Device Controller Not Ready). Этот бит указывает, что система находится в процессе завершения переключений состояния после выхода из гибернации. Требуется 256 тактов синхросигнала шины с момента установки DCTL[31].Run/Stop, чтобы переключение состояний завершилось. Во время гибернации, если PHY UTMI/ULPI приостановлена, тогда задержка в 256 тактов синхросигнала шины начинается после того,	0x0	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
		как PHY выйдет из состояния приостановки. Программное обеспечение должно устанавливать 1 в DCTL[31].Run/Stop и ожидать до тех пор, пока этот бит будет снят (обнулен), чтобы присутпить к обработке DSTS.USBLnkSt. Этот бит валиден только когда в DWC_USB3_EN_PWROPT установлено 2 и GCTL[1].GblHibernationEn=1. Контролепригодность: writeAsRead. Изменчивость: true. Ограничение записи: writeAsRead.		
[31:30]	reserved_31_30	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0x3. Изменчивость: true. Ограничение записи: writeAsRead.	0x0	R

26.4.5 Регистр параметра универсальной команды устройства (DGCMDPAR)

Этот регистр указывает параметр команды устройства. Он должен быть запрограммирован заранее или вместе с командой устройства. Доступные команды устройства перечислены в регистре DGCMD.

Поля регистра DGCMDPAR описаны в Таблица 26.72.

Таблица 26.72. Регистр параметра универсальной команды устройства

Диапазон	Название	Описание	Исходное значение	Тип доступа
[31:0]	PARAMETER	PARAMETER.	0x0	R/W

26.4.6 Регистр универсальной команды устройства (DGCMD)

Этот регистр позволяет программному обеспечению программировать систему, используя интерфейс, состоящий из одиночной универсальной команды для отправки пакетов и уведомлений по организации канала. Этот регистр содержит поля команды, управления и статуса, относящиеся к текущей универсальной команде, в то время как регистр DGCMDPAR обеспечивает параметр команды.

Поля регистра DGCMD описаны в Таблица 26.73.

Таблица 26.73. Регистр универсальной команды устройства

Диапазон	Название	Описание	Исходное значение	Тип доступа
[7:0]	CMDTYP	Тип универсальной команды (Generic Command Type).	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Задает тип универсальной команды, запрос на выполнение которой драйвер программного обеспечения отправляет контроллеру. 02h: установить периодические параметры (Set Periodic Parameters); 04h: установить младшие биты адреса массива буфера блокнотной памяти (Set Scratchpad Buffer Array Address Lo); 05h: установить старшие биты адреса массива буфера блокнотной памяти (Set Scratchpad Buffer Array Address Hi); 07h: передать уведомление устройства (Transmit Device Notification); 09h: очистка выбранного буфера FIFO (Selected FIFO Flush); 0Ah: очистка всех буферов FIFO (All FIFO Flush); 0Ch: установить состояние NRDY (неготовности) конечной точки (Set Endpoint NRDY); 10h: провести тест обхода шины SoC (Run SoC Bus LoopBack Test); 11h: перезапустить после отсоединения (Restart After Disconnect). Все остальные значения зарезервированы.		
[8]	CMDIOC	Прерывание команды по завершении (Command Interrupt on Complete). Когда этот бит установлен, контроллер устройства отправляет событие универсального завершения команды (Generic Command Completion) после исполнения команды. Следует отметить, что это прерывание поставлено в соответствие DCFG.IntrNum. Замечание: в это поле не следует устанавливать 1, если в поле DCTL.RunStop установлен 0.	0x0	R/W
[9]	reserved_9	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0x1. Ограничение записи: writeAsRead.	0x0	R
[10]	CMDACT	Команда активна (Command Active). Программное обеспечение устанавливает 1 в этот бит, чтобы позволить контроллеру устройства исполнить универсальную команду. Контроллер устройства устанавливает в этот бит 0 после исполнения команды. Изменчивость: true.	0x0	R/W
[11]	reserved_11	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0x1. Ограничение записи: writeAsRead.	0x0	R
[15:12]	CMDSTATUS	Статус команды (Command Status).	0x0	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
		1: CmdErr: указывает, что контроллер устройства столкнулся с ошибкой при обработке команды; 0: указывает, что исполнение команды успешно. Контролепригодность: writeAsRead. Ограничение записи: writeAsRead.		
[31:16]	reserved_31_16	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0xffff. Ограничение записи: writeAsRead.	0x0	R

26.4.7 Регистр подключения активных конечных точек USB устройства (DALEPENA)

Этот регистр указывает, активна ли конечная точка USB в данной конфигурации или интерфейсе.

Поля регистра DALEPENA описаны в Таблица 26.74.

Таблица 26.74. Регистр подключения активных конечных точек USB устройства

Диапазон	Название	Описание	Исходное значение	Тип доступа
[31:0]	USBACTEP	Активные конечные точки USB (USB Active Endpoints (USBActEP)). Это поле указывает, активна ли конечная точка USB в текущей конфигурации и интерфейсе. Это применимо к конечным точкам IN 0.15 и конечным точкам OUT 0.15 USB, с одним битом для каждой из 32 возможных конечных точек. Четные номера предназначены для конечных точек OUT USB, а нечетные номера предназначены для конечных точек IN USB, следующим образом: бит [0]: USB EP0-OUT; бит [1]: USB EP0-IN; бит [2]: USB EP1-OUT; бит [3]: USB EP1-IN. Модуль, программирующий этот регистр, должен установить биты 0 и 1, поскольку они подключают управляющие конечные точки (control endpoints), которые соответствуют физическим конечным точкам (ресурсам) после USBReset. Аппаратное обеспечение очищает эти биты для всех конечных точек (кроме EP0-OUT и EP0-IN) после обнаружения события сброса USB. После получения запросов SetConfiguration и SetInterface, приложение должно запрограммировать регистры конечных точек соответственно и установить эти биты. Более подробно это описано в Разделе “Flexible Endpoint Mapping” (Гибкое обеспечение соответствия конечных точек) в справочнике сверхскоростного контроллера USB 3.0 от DesignWare Cores.	0x0	R/W

26.4.8 Зарезервированный регистр (Rsvd[0:31])

Смещение: 0xc724 + (i * 0x4), где i = 0..31.

Поля регистра Rsvd[0:31] описаны в Таблица 26.75.

Таблица 26.75. Зарезервированный регистр

Диапазон	Название	Описание	Исходное значение	Тип доступа
[31:0]	reserved_31_0	Зарезервировано. Контролепригодность: writeAsRead. Маска сброса: 0xffffffff. Ограничение записи: writeAsRead.	0x0	R

26.4.9 Регистр параметра 2 команды физической конечной точки-n устройства (DEPCMDPAR2[0:7])

Этот регистр указывает параметр 2 команды физической конечной точки. Перед отправкой команды он должен быть запрограммирован.

Смещение: 0xc800 + (i * 0x10), где i = 0..=[eval_param DWC_USB3_NUM_EPS]-1.

Поля регистра DEPCMDPAR2[0:7] описаны в Таблица 26.76.

Таблица 26.76. Регистр параметра 2 команды физической конечной точки-n устройства

Диапазон	Название	Описание	Исходное значение	Тип доступа
[31:0]	PARAMETER	PARAMETER. Контролепригодность: нетестируемо. Маска сброса: 0xffffffff. Ограничение записи: writeAsRead.	0x0	R/W

26.4.10 Регистр параметра 1 команды физической конечной точки-n устройства (DEPCMDPAR1[0:7])

Смещение: 0xc804 + (i * 0x10), где i = 0..=[eval_param DWC_USB3_NUM_EPS]-1.

Поля регистра DEPCMDPAR1[0:7] описаны в Таблица 26.77.

Таблица 26.77. Регистр параметра 1 команды физической конечной точки-n устройства

Диапазон	Название	Описание	Исходное значение	Тип доступа
[31:0]	PARAMETER	PARAMETER. Контролепригодность: нетестируемо. Маска сброса: 0xffffffff. Ограничение записи: writeAsRead.	0x0	R/W

26.4.11 Регистр параметра 0 команды физической конечной точки-n устройства (DEPCMDPAR0[0:7])

Смещение: 0xc808 + (i * 0x10), где i = 0..=[eval_param DWC_USB3_NUM_EPS]-1.

Поля регистра DEPCMDPAR0[0:7] описаны в Таблица 26.78.

Таблица 26.78. Регистр параметра 0 команды физической конечной точки-n устройства

Диапазон	Название	Описание	Исходное значение	Тип доступа
[31:0]	PARAMETER	PARAMETER. Контролепригодность: нетестируемо. Маска сброса: 0xffffffff. Ограничение записи: writeAsRead.	0x0	R/W

26.4.12 Регистр команд физической конечной точки-n устройства (DEPCMD[0:7])

Этот регистр позволяет программному обеспечению отправлять команды, специфичные для конечных точек. Этот регистр содержит поля команд, управления и статуса, соответствующие текущей универсальной команде, в то время как регистры DEPCMDPAR[2:0]n обеспечивают параметры команд и возвращают информацию по статусу.

Несколько полей (включая поле «Тип команды (Command Type)») имеют доступ только на запись, так что их значения при чтении неопределены. После подключения питания, перед отправкой первой команды конечной точке, значение при чтении этого регистра неопределено. В частности, бит CmdAct может быть установлен после подключения питания. В таком случае отправлять команду к конечной точке безопасно.

Смещение: 0xc80c + (i * 0x10), где i = 0..=[eval_param DWC_USB3_NUM_EPS]-1.

Поля регистра DEPCMD[0:7] описаны в Таблица 26.79.

Таблица 26.79. Регистр команд физической конечной точки-n устройства

Диапазон	Название	Описание	Исходное значение	Тип доступа
[3:0]	CMDTYP	Тип команды (Command Type). Задает тип команды, запрос на выполнение которой драйвер программного обеспечения отправляет системе. 00h: зарезервировано; 01h: установка конфигурации конечной точки – 64 или 96-битный параметр; 02h: установка конфигурации передачи ресурса конечной точке – 32-битный параметр; 03h: получение состояния конечной точки – не требуется параметра; 04h: установка остановки (Set Stall) – не требуется параметра; 05h: очистка остановки (см. установка остановки) – не требуется параметра; 06h: начало передачи – 64-битный параметр; 07h: обновление передачи – не требуется параметра; 08h: окончание передачи – не требуется параметра; 09h: запуск новой конфигурации – не требуется параметра; Контролепригодность: нетестируемо. Маска сброса: 0xf. Ограничение записи: writeAsRead.	0x0	R/W
[7:4]	reserved_7_4	Зарезервировано. Контролепригодность: нетестируемо. Маска сброса: 0xf. Ограничение записи: writeAsRead.	0x0	R
[8]	CMDIOC	Прерывание команды по завершении (CmdIOC). Когда этот бит установлен, контроллер устройства отправляет универсальное событие завершения команды конечной точки (Endpoint Command Complete) после исполнения команды. Следует отметить, что это прерывание поставлено в соответствие DEPCFG.IntrNum. Когда исполняется команда DEPCFG, прерывание команды по завершении отправляется к прерыванию, указанному DEPCFG.IntrNum в текущей команде. Замечание: в это поле не следует устанавливать 1, если поле DCTL.RunStop имеет значение 0. Контролепригодность: нетестируемо. Маска сброса: 0x1. Ограничение записи: writeAsRead.	0x0	R/W
[9]	reserved_9	Зарезервировано.	0x0	R

Диапазон	Название	Описание	Исходное значение	Тип доступа
		Контролепригодность: нетестируемо. Маска сброса: 0x1. Ограничение записи: writeAsRead.		
[10]	CMDACT	Команда активна (Command Active). Программное обеспечение устанавливает 1 в этот бит для того, чтобы позволить контроллеру конечной точки устройства исполнить универсальную команду. Контроллер устройства устанавливает 0 в этот бит, когда поле CmdStatus валидно и конечная точка готова принять следующую команду. Это не подразумевает, что все эффекты команды, отправленной ранее, уже осуществлены. Контролепригодность: нетестируемо. Маска сброса: 0x1. Ограничение записи: writeAsRead.	0x0	R/W
[11]	HIPRI_FORCERM	HighPriority/ForceRM. HighPriority: валидно только для команды запуска передачи (Start Transfer). ForceRM: валидно только для команды окончания передачи (End Transfer). ClearPendIN: валидно только для команды очистки остановки (Clear Stall). Программное обеспечение устанавливает этот бит, чтобы очистить любые транзакции IN, ожидающие обработки (на данной конечной точке), застрявшие на нижних уровнях, когда отправлена команда очистки остановки (Clear Stall). Контролепригодность: нетестируемо. Маска сброса: 0x1. Ограничение записи: writeAsRead.	0x0	R/W
[15:12]	CMDSTATUS	Статус завершения команды (Command Completion Status). Добавочная информация о завершении команды доступна в этом поле. Информация представлена в том же формате, что и биты 15:12 события завершения команды конечной точки (Endpoint Command Complete). Подробнее см. таблицу «События конечной точки-n устройства: DEPEVT» (Device Endpoint-n Events: DEPEVT) в справочнике. Контролепригодность: нетестируемо. Маска сброса: 0xf. Ограничение записи: writeAsRead.	0x0	R/W
[31:16]	COMMANDPARAM	Параметры команды или события (Command Parameters or Event Parameters). Когда в этот регистр производится запись, в этих битах хранятся параметры команды (CommandParam):	0x0	R/W

Диапазон	Название	Описание	Исходное значение	Тип доступа
		– для команды запуска передачи (Start Transfer): [31:16]: идентификатор потока (StreamID). Идентификатор потока USB, присвоенный этой передаче. – для команды запуска передачи (Start Transfer), отправленной к изохронной конечной точке: [31:16]: StartMicroFramNum: указывает номер (микро)фрейма, к которому относится первый блок TRB. – для команд передачи обновления (Update Transfer), конца передачи (End Transfer), и начала новой конфигурации (Start New Configuration): [22:16]: индекс передачи ресурса (Transfer Resource Index) (XferRscIdx). Индекс передачи ресурса, присвоенный аппаратно, который возвращается в ответ на команду запуска передачи. Индекс передачи ресурса, присвоенный программным обеспечением приложения, для команды запуска новой конфигурации (Start New Configuration). Когда происходит чтение данного регистра, в этих битах хранятся параметры событий (Event Parameters (EventParam)). Подробнее эти биты описаны в таблице “Device Endpoint-n Events: DEPEVT” (события конечной точки n устройства: DEPEVT) справочника. Контролепригодность: нетестируемо. Маска сброса: 0xffff. Ограничение записи: writeAsRead.		

26.4.13 Регистр модерации прерываний устройства (DEV_IMOD[0])

Этот регистр управляет функцией модерации прерывания, которая позволяет программному обеспечению устройства регулировать (throttle) объем прерываний.

Ключевые функции:

- модерация прерывания подключается только когда в интервал IMOD запрограммирована ненулевая величина;
- прерывание устанавливается всегда, когда счётчик IMOD (обратного отсчёта) равен 0, EVNT_HANDLER_BUSY равен 0 и присутствуют события, ожидающие обработки (то есть, счётчик событий имеет ненулевое значение);

- когда устанавливается прерывание, GEVNTCOUNT[EVNT_HANDLER_BUSY] устанавливается аппаратно и очищается программным обеспечением, когда обработка прерывания завершается;
- линия прерывания (Interrupt line) снимается после первой записи в счетчик событий;
- всякий раз, когда линия прерывания снимается, в счетчик IMOD загружается интервал IMOD.

Смещение: 0xca00 + (i * 0x4), где i = 0..=[eval_param DWC_USB3_DEVICE_NUM_INT]-1.

Поля регистра DEV_IMOD[0] описаны в Таблица 26.80.

Таблица 26.80. Регистр модерации прерываний устройства

Диапазон	Название	Описание	Исходное значение	Тип доступа
[15:0]	DEVICE_IMODI	Интервал модерации (Moderation Interval (DEVICE_IMODI)) Это поле содержит минимальный интервал прерываний между событиями. Интервал задается в величинах инкремента в 250 нс. Значение 0 отключает логику регулирования прерывания и прерывания генерируются незамедлительно если счетчик событий приобретает ненулевое значение. В режиме симуляции уменьшения масштаба (scaledown simulation mode) для отсчёта 250 нс используются 4 такта ram. Маска сброса: 0xffff. Изменчивость: true.	0x0	R/W
[31:16]	DEVICE_IMODC	Обратный счетчик модерации прерываний (Interrupt Moderation Down Counter). В этот счетчик загружено значение DEVICE_IMODI, и когда аппаратная линия прерывания(n) переходит из состояния «установлено» в состояние «снято», этот счетчик выполняет обратный отсчет до 0 и останавливается. Сигнал прерывания(n) отправляется всегда, когда этот счетчик имеет значение 0, EVNT_HANDLER_BUSY имеет значение 0, и присутствуют события, ожидающие обработки (то есть, счетчик событий имеет ненулевое значение). Программное обеспечение может писать в этот счетчик напрямую в любое время, чтобы модифицировать объем прерываний. Маска сброса: 0xffff. Изменчивость: true.	0x0	R/W

27. КОНТРОЛЛЕР ETHERNET (EMAC)

27.1 Общая информация

Контроллер GEMAC обеспечивает передачу данных в соответствии с протоколом Ethernet/IEEE 802.3-2012 на скоростях 10/100/1000 Мбит/с с аппаратно обеспечиваемой поддержкой протокола IEEE-1588. Ядро GEMAC может работать на скоростях передачи: 10 Мбит/с, 100 Мбит/с (быстрый Ethernet) и 1000 Мбит/с (1-гигабитный Ethernet). Со стороны системы в ядре GEMAC реализован 64-битный AXI Master-интерфейс для передачи пакетов между памятью и встроенным FIFO (под управлением Scatter-Gather DMA) и 32-битный AXI Slave-интерфейса для доступа к конфигурационным регистрам блока. В контроллере GEMAC используются встроенные FIFO как со стороны передачи, так и со стороны приема, для передачи и получения фреймов.

Контроллер GEMAC поддерживает полудуплексный режим работы для скоростей 10/100 Мбит/с, и полнодуплексный режим работы для скоростей 10/100/1000 Мбит/с. При работе в полудуплексном режиме, контроллер GEMAC полностью совместим с разделом 4 ISO/IEC 8802-3 (ANSI/IEEE стандарта) и с ANSI/IEEE 802.3. При работе в полнодуплексном режиме, контроллер GEMAC совместим со стандартом IEEE 802.3x для полнодуплексных операций, и также с IEEE 802.3Qbb для поддержки приоритетных фреймов-пауз.

В ядре GEMAC обеспечены расширенные программируемые функции для минимизации сложности пост/пред обработки сообщений. Эти функции включают динамическую пофреймовую генерацию FSC, автоматическое добавление полей для обеспечения минимального размера фрейма, автоматическая переотправка и обнаружение конфликтов фреймов (в полудуплексном режиме), проверка FCS входящих фреймов, дополнительную разборку поля FCS, автоматическое обнаружение фреймов-пауз (PAUSE frame) и приоритетных фреймов управления паузами (Priority PAUSE Control frame) при работе в полнодуплексном режиме, обеспечение фильтрации адресов при получении фреймов и т.д.

В контроллере GEMAC реализована поддержка протокола IEEE-1588 (как V1, так и V2), обеспечиваемая аппаратно при работе в полнодуплексном режиме на скоростях 10/100/1000 Мбит/с. Поддерживается работа системного таймера, обеспечен программный интерфейс для настройки системного таймера. Также организована поддержка проставления временных меток на полученные и передаваемые пакеты.

Встроенный контроллер DMA (scater-gater) реализует прямой доступ в памяти с программируемым размером транзакции, обеспечивающим минимальное использование ресурсов CPU. Контроллер DMA передает фреймы из/во внутренние буферы FIFO и буферы данных в память хоста. Дескрипторы, которые остаются в памяти хоста, работают как указатели на эти буферы. В контроллере DMA операции передачи и получения поддерживаются независимо. Контроллер DMA управляется набором DMA регистров.

Контроллер GEMAC может поддерживать передачу пакетов минимального размера вплотную друг за другом на полной скорости передачи канала с межпакетным зазором (IPG) размером в 96 интервалов передачи бита, эквивалентные 90,6 мкс для скорости 10 Мбит/с, 0.96 мкс для 100 Мбит/с, и 0.096 мкс для 1000 Мбит/с. С получающей стороны контроллер GEMAC может обработать фреймы минимального размера, идущие друг за другом вплотную на полной скорости передачи канала с IPG в 64 интервала передачи бита.

В основном контроллер GEMAC исполняет следующие функции:

- передача и получение сообщений пакетированных данных с фреймингом (определение границ фреймов, синхронизация фреймов) и обнаружение ошибок (ошибки передачи, связанные с физической средой).
- управление распределением медиаданных (для предупреждения столкновений) и разрешение конфликтов (обработка столкновений) в полудуплексном режиме работы на скоростях 10 или 100 Мбит/с.
- повторная отправка конфликтующих фреймов в полудуплексном режиме.
- поддержка функций управления потоком (Flow Control) в полнодуплексном режиме путем декодирования фреймов управляющих паузами и отключения передатчика и формирование пакетов управляющих паузами.
- поддержка RGMII (сокращенный GMII) интерфейса для сокращения числа контактов.
- поддержка интерфейса MDC/MDIO для управления внешним PHY.
- использование 64-разрядного AXI интерфейса для передачи пакетов между памятью хоста и внутренними буферами FIFO.
- автоматическая передача пакетов между памятью хоста и внутренними буферами FIFO (на основе дескрипторов) для снижения нагрузки на CPU.
- выставление временных меток на пакеты для поддержки IEEE-1588 (PTP).

27.1.1 Основные функции

Ядро GEMAC полностью совместимо со спецификацией IEEE 802.3-2012 и остальными дополнительными спецификациями по функционалу, и поддерживает следующие функции:

- скорость передачи данных 10/100/1000 Мбит/с
- независимый 64-битный scatter-gather DMA для операций передачи и приема.
- RGMII для связи с внешним PHY
- поддержка VLAN Q-Tag frame
- Full-Duplex режим для скоростей 10/100/1000 Мбит/с и Half-Duplex режим для 10/100 Мбит/с
- протокол CSMA/CD для Half-Duplex режима с обнаружением и предотвращением коллизий
- автоматическая ретрансляция кадров при коллизии в полудуплексном режиме
- управление потоком на основе фреймов «PAUSE» для Full-Duplex режима
- управление потоком через «Priority PAUSE» на базе стандарта IEEE 802.3Qbb
- генерация фреймов управления на интерфейсе MDC/MDIO под контролем программного обеспечения для взаимодействия с внешним устройством PHY
- совместимая с 802.3 поддержка управления MIB, SNMP, RMON путем использования набора счетчиков.
- поддержка «Jumbo» фреймов как для операций приема, так и передачи.
- поддержка IEEE-1588 (V1 и V2) путем внедрения системного таймера и проставления временных меток (Timestamping) на отправленные и полученные пакеты под контролем программного обеспечения
- управление питанием благодаря поддержке Magic Packet и Wake-Up Frame

27.1.2 Функции DMA

- 64-разрядный scatter-gather DMA с независимыми каналами записи и чтения данных
- возможность использования дескрипторов в виде связанного списка или кольцевых цепочек

- программируемая длина пропуска между дескрипторами в случае цепочки дескрипторов.
- возможность использования до двух буферов на дескриптор
- программируемый размер буфера (до 1 Кбайт) настраиваемый для каждого дескриптора.
- программируемый размер пакета для эффективного использования хост-шины
- поддержка выравнивания буфера на байтовом уровне для буферов передачи и приема
- гибкая система арбитража передачи/приема
- программируемая структура запросов на прерывания
- операции с прямым и обратным порядком следования байтов (Big/Little Endian) как для буферов данных, так и для структур дескрипторов

27.1.3 Функции передатчика

- переменная длина (64 BT (битовых интервала), 96 BT, 128 BT, 256 BT) межфреймового интервала (IFG) при передаче между кадрами, со значением по умолчанию 128 битовых интервалов IFG.
- переменная длина (3, 5, 7 байтов) преамбулы со значением по умолчанию 7 байт
- автоматическая генерация FCS во время передачи
- автоматическая генерация PAD во время передачи для удовлетворения требования минимального размера фрейма (MinFrameSize)
- возможность отключить вставку PAD и/или CRC32 при передаче для каждого кадра
- возможность защиты приложения для организации процесса переотправки фреймов при конфликтах во время работы в полудуплексном режиме.

27.1.4 Функции приемника

- проверка на минимальный межфреймовый интервал (IFG) между фреймами. Минимальный обрабатываемый IFG во время получения фреймов вплотную друг за другом – 64 битовых интервала;

- обнаружение и отбрасывание преамбулы при приеме. Обработка принятых кадров без преамбулы и только SFD
- гибкие режимы фильтрации адресов
    - четыре 48-битных MAC-адреса для идеального совпадения адреса с функцией подключения/отключения индивидуального адреса;
    - возможность обратной фильтрации адресов для четырех вышеописанных 48-битных MAC-адресов;
    - 64-битная хеш-таблица для фильтрации групповых адресов;
    - режим приема всех пакетов независимо от того, кому они предназначены;
    - прием широковещательных пакетов
- автоматическая проверка поля FCS на правильность значения CRC
- автоматическая проверка карликовых фреймов (Runt frames) с возможностью отфильтровать их из приложения
- автоматическая проверка длины поля DATA в случае кадров типа 802.3 с полем длины
- настраиваемое поле для обнаружения фреймов MaxFrameLen
- автоматическая настройка поля MaxFrameLen для фреймов, тегированных VLAN;
- полная информация о 32-битном статусе предоставляется для каждого принятого кадра

27.1.5 Функции управления потоком:

- Программно управляемая генерация кадра PAUSE с программируемыми задержками
- Программно управляемая генерация кадра “Priority Pause” с программируемыми задержками паузы на основе приоритета
- Формирование управляющих кадров PAUSE на основе событий “FIFO Almost-Full” и “FIFO Almost-Empty”
- Формирование кадра “Priority Pause” (до 8 приоритетов) под управлением приложения или под управлением программного обеспечения

- Автоматическая повторная передача кадров “Pause”/“Priority Pause”, если внутренний таймер паузы достигает половины и условие паузы все еще существует
- Возможность использовать зарезервированный групповой адрес рассылки или запрограммированный уникальный адрес рассылки в поле DA кадра “PAUSE”
- Генерация кадра “PAUSE”, даже когда логика передачи находится в режиме паузы
- Автоматическое обнаружение кадров “PAUSE” с полем DA либо зарезервированного группового адреса, либо уникального MAC-адреса устройства
- Автоматическое определение кадров “Priority PAUSE” с полем DA либо зарезервированного группового адреса, либо уникального MAC-адреса устройства
- Проверка корректности OPCODE, размера кадра и поля FCS в кадрах “PAUSE”
- Отключение передатчика на время, указанное в поле паузы в полученном кадре “PAUSE”

27.1.6 Функции IEEE 1588:

- Определение пакетов, которые требуют метки времени в направлении передачи (из дескрипторов)
- хранение значения метки времени передачи для ПО
- Анализ входящих пакетов и идентификация PTP пакетов (протокола «точного времени»):
    - поддержка форматов фреймов IEEE 1588 V1 и V2;
    - поддержка пакетов V1 PTP в виде полезной нагрузки UDP при версии IPV4;
    - поддержка пакетов V2 PTP либо в виде специального Ethertype, либо в виде полезной нагрузки протокола UDP при версиях IPV4 или IPV6.
- Идентификация входящих пакетов требующих метки времени (на основе идентификатора сообщения PTP)
- Передача PTP пакета принятого фрейма и индикация меток времени программному обеспечению
- Хранение значения полученной метки времени для ПО
- Реализация системного таймера с программируемым значением приращения и периодом приращения

- Возможность доступа ПО к настройкам системного таймера
- Реализация логики 1588 в отдельном домене синхронизации, независимом от частот приема/передачи и тактовых частот хоста
- Никакие модификации пакетов не выполняются ни в режиме передачи, ни в режиме приема, когда включена двухэтапная временная метка.

27.2 Структурная схема

На Рисунок 27.1 показана блок-схема высокого уровня вместе с различными функциональными блоками контроллера GEMAC с поддержкой IEEE 1588.

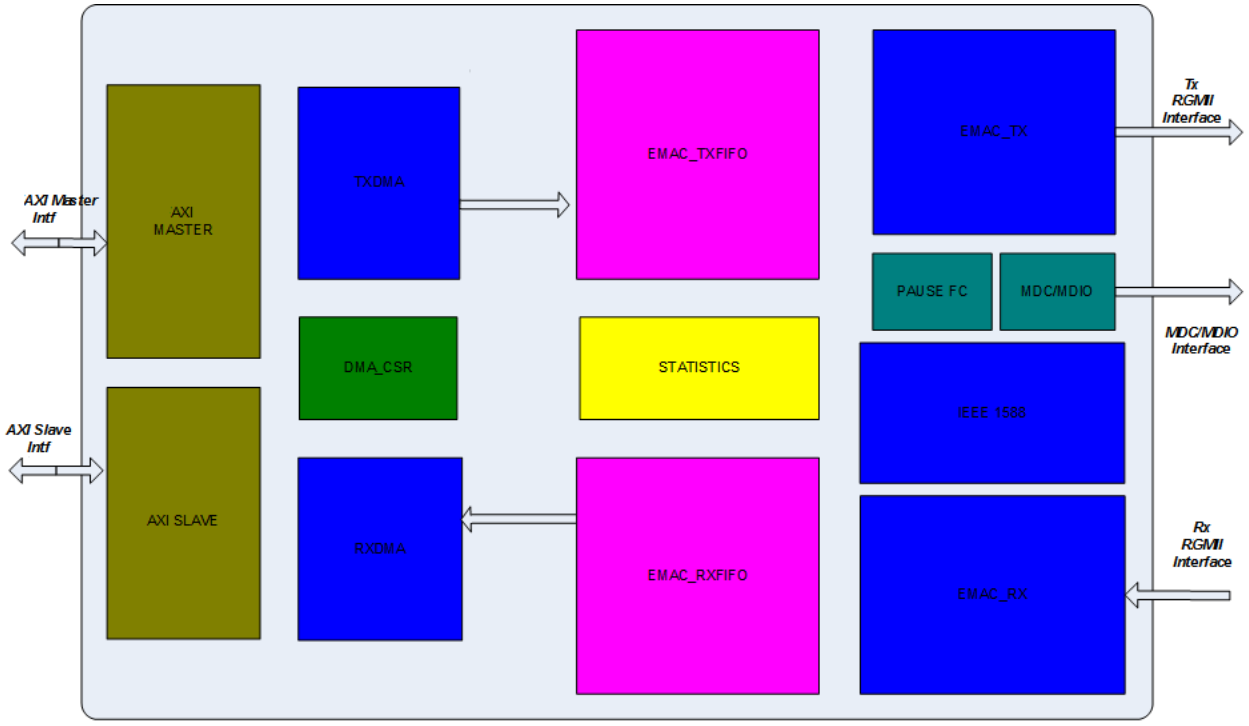


Рисунок 27.1. Блок-схема контроллера GEMAC

27.2.1 Описание подблоков GEMAC

Ниже приводится краткое описание различных модулей контроллера GEMAC:

- EMAC_TX: содержит конечный автомат, формирующий (в соответствии с протоколом CSMA/CD) передающийся кадр и с заданной скоростью транслирующий его на RGMII интерфейс.
- EMAC_RX: содержит конечный автомат, обрабатывающий (в соответствии с протоколом CSMA/CD) принимающийся по RGMII интерфейсу кадр, а также выполняющий разбивку кадра по полям и проверку ошибок.
- IEEE_1588: реализует поддержку протокола IEEE 1588, выполняя проставку временных меток входящих и выходящих PTP пакетов под управлением программного обеспечения. Также содержит конфигурируемый системный таймер с возможностью программной подстройки.
- PAUSE_FC: декодирует полученные пакеты PAUSE (Pri-Pause), если GEMAC контроллер находится в Full Duplex режиме, а также содержит PAUSE_TIMER, позволяющий отключить EMAC_TX модуль. Также этот модуль по программному запросу генерирует пакеты PAUSE (Pri-Pause).
- MDC/MDIO Controller: формирует MDIO пакеты для общения с внешними PHY

- STATISTICS: содержит различные счетчики, собирающие статистику операций передачи и приема
- EMAC_RXFIFO: содержит FIFO (8KB), предназначенное для хранения принятых кадров до того, как они будут перемещены во внешнюю память.
- EMAC_TXFIFO: содержит FIFO (8KB), предназначенное для хранения кадров полученных из памяти прежде, чем они будут переданы на интерфейс Ethernet
- EMAC_WKUP: осуществляет вывод блока из состояния ожидания посредством Magic Packet. В этом режиме, когда получен Magic Packet, генерируется отдельный сигнал к ведущему интерфейсу
- TXDMA: осуществляет передачу данных из внешней памяти в Transmit FIFO (EMAC_TXFIFO), используя дескриптор передачи.
- RXDMA: осуществляет передачу данных из Receive FIFO (EMAC_RXFIFO) во внешнюю память, используя дескриптор приема.
- Registers: В этом модуле находятся регистры управления и статуса, позволяющие управлять операциями принимающего и передающего DMA, операциями передачи и приема Ethernet интерфейса. Этот модуль также обеспечивает доступ к регистрам статистики и статуса прерываний.
- AXI Master Interface: обеспечивает функциональность AXI Master интерфейса генерирующего транзакции на шине AXI на основе запросов от RXDMA/TXDMA.
- AXI Slave Interface: обеспечивает функциональность AXI Slave. Данный интерфейс используется для доступа ко всем регистрам DMA/MAC.

27.3 Регистры

Регистры GEMAC используются для управления работой контроллера, а также для считывания статуса/прерывания.

Регистры – 32-битные выровненные слова, и обращения к ним должны осуществляться исключительно с использованием выровненных по 32-нох границе адресов. В зарезервированных полях должен быть записан 0, и контроллер также возвращает значение 0 в этих полях.

В момент сброса во всех регистрах устанавливаются их значения по умолчанию.

Существует три набора регистров контроллера GEMAC:

- регистры управления работой контроллера DMA (начиная со смещения 0x0000);
- регистры управления работой MAC-интерфейса (начиная со смещения 0x0100);
- регистры управления работой PTP (начиная с 0x0300);

Перечень регистров приведен в Таблица 27.1.

Таблица 27.1. Перечень программно-доступных регистров

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
DmaCfg	DMA_Configuration_Register	0x40	RW	0x0000
DmaCtl	DMA_Control_Register	0x0	RW	0x0004
DmaSts	DMA_Status_and_IRQ_Register	0x0	RW	0x0008
DmaIntrEn	DMA_Interrupt_Enable_Register	0x0	RW	0x000C
DmaTxAutoPollCnt	DMA_Transmit_Auto_Poll_Counter_Register	0x0	RW	0x0010
DmaTxPollDemand	DMA_Transmit_Poll_Demand_Register	0x0	WO	0x0014
DmaRxPollDemand	DMA_Receive_Poll_Demand_Register	0x0	WO	0x0018
DmaTxBaseAddr	DMA_Transmit_Base_Address_Register	0x0	RW	0x001C
DmaRxBaseAddr	DMA_Receive_Base_Address_Register	0x0	RW	0x0020
DmaMissFrameCounter	DMA_Missed_Frame_Counter_Register	0x0	RO	0x0024
DmaStopFlushCounter	DMA_Stop_Flush_Counter_Register	0x0	RO	0x0028
DmaRxIntrMitigationCtl	DMA_Receive_Interrupt_Mitigation_Control	0xFFFF01	RW	0x002C
DmaCurTxDescrPointer	DMA_Current_Transmit_Descriptor_Pointer_Register	0x0	RO	0x0030
DmaCurTxBufPointer	DMA_Current_Transmit_Buffer_Pointer_Register	0x0	RO	0x0034
DmaCurRxDescrPointer	DMA_Current_Receive_Descriptor_Pointer_Register	0x0	RO	0x0038
DmaCurRxBufPointer	DMA_Current_Receive_Buffer_Pointer_Register	0x0	RO	0x003C
MacGlobalCtl	MAC_Global_Control_Register	0x0	RW	0x0100
MacTxCtl	MAC_Transmit_Control_Register	0x0	RW	0x0104
MacRxCtl	MAC_Receive_Control_Register	0x0	RW	0x0108
MacMaxFrameSize	MAC_Maximum_Frame_Size_Register	0x5EE	RW	0x010C
MacTxJabberSize	MAC_Transmit_Jabber_Size_Register	0x600	RW	0x0110

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
MacRxJabberSize	MAC_Receive_Jabber_Size_Register	0x600	RW	0x0114
MacAddrCtl	MAC_Address_Control_Register	0x0	RW	0x0118
MacMDIOClockDivCtl	MAC_MDIO_Clock_Division_Control_Register	0x40	RW	0x011C
MacAddr1Hi	MAC_Address1_High_Register	0x0	RW	0x0120
MacAddr1Med	MAC_Address1_Med_Register	0x0	RW	0x0124
MacAddr1Lo	MAC_Address1_Low_Register	0x0	RW	0x0128
MacAddr2Hi	MAC_Address2_High_Register	0x0	RW	0x012C
MacAddr2Med	MAC_Address2_Med_Register	0x0	RW	0x0130
MacAddr2Lo	MAC_Address2_Low_Register	0x0	RW	0x0134
MacAddr3Hi	MAC_Address3_High_Register	0x0	RW	0x0138
MacAddr3Med	MAC_Address3_Med_Register	0x0	RW	0x013C
MacAddr3Lo	MAC_Address3_Low_Register	0x0	RW	0x0140
MacAddr4Hi	MAC_Address4_High_Register	0x0	RW	0x0144
MacAddr4Med	MAC_Address4_Med_Register	0x0	RW	0x0148
MacAddr4Lo	MAC_Address4_Low_Register	0x0	RW	0x014C
MacHashTab1	MAC_Hash_Table1_Register	0x0	RW	0x0150
MacHashTab2	MAC_Hash_Table2_Register	0x0	RW	0x0154
MacHashTab3	MAC_Hash_Table3_Register	0x0	RW	0x0158
MacHashTab4	MAC_Hash_Table4_Register	0x0	RW	0x015C
MacFlowControlCtl	MAC_FlowControl_Control_Register	0x0	RW	0x0160
MacFlowControlPriPauseFrameGen	MAC_FlowControl_PriPause_Frame_Generate_Register	0x0	RW	0x0164
MacFlowControlSrcAddrHi	MAC_FlowControl_Source_Address_High_Register	0x0	RW	0x0168
MacFlowControlSrcAddrMed	MAC_FlowControl_Source_Address_Med_Register	0x0	RW	0x016C
MacFlowControlSrcAddrLo	MAC_FlowControl_Source_Address_Low_Register	0x0	RW	0x0170
MacFlowControlDstAddrHi	MAC_FlowControl_Destination_Address_High_Register	0x0	RW	0x0174
MacFlowControlDstAddrMed	MAC_FlowControl_Destination_Address_Med_Register	0x0	RW	0x0178
MacFlowControlDstAddrLo	MAC_FlowControl_Destination_Address_Low_Register	0x0	RW	0x017C
MacFlowControlPauseTimeValue	MAC_FlowControl_Pause_Time_Value_Register	0x0	RW	0x0180
MacFlowControlAutoGenHiPauseTimeValue	MAC_FlowControl_Auto_Gen_Hi_Pause_Time_Value_Register	0x0	RW	0x0184
MacFlowControlAutoGenLoPauseTimeValue	MAC_FlowControl_Auto_Gen_Lo_Pause_Time_Value_Register	0x0	RW	0x0188
MacFlowControlAutoPauseFrameGenHiThresh	MAC_Flow_Control_Auto_Pause_Frame_Gen_Hi_Threshold_Register	0xFFFF	RW	0x018C
MacFlowControlAutoPauseFrameGenLowThresh	MAC_Flow_Control_Auto_Pause_Frame_Gen_Low_Threshold_Register	0x0	RW	0x0190
MacMDIOCtl	MAC_MDIO_Control_Register	0x0	RW	0x01A0
MacMDIOData	MAC_MDIO_Data_Register	0x0	RW	0x01A4
MacRxStatCtrCtl	MAC_Rx_StatCtr_Control_Register	0x0	RW	0x01A8
MacRxStatCtrDataHi	MAC_Rx_StatCtr_Data_High_Register	0x0	RO	0x01AC
MacRxStatCtrDataLo	MAC_Rx_StatCtr_Data_Low_Register	0x0	RO	0x01B0
MacTxStatCtrCtl	MAC_Tx_StatCtr_Control_Register	0x0	RW	0x01B4
MacTxStatCtrDataHi	MAC_Tx_StatCtr_Data_High_Register	0x0	RO	0x01B8
MacTxStatCtrDataLo	MAC_Tx_StatCtr_Data_Low_Register	0x0	RO	0x01BC
MacTxFIFOAlmostFull	MAC_Transmit_FIFO_Almost_Full_Register	0x0	RW	0x01C0

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
MacTxPacketStartThresh	MAC_Transmit_Packet_Start_Threshold_Register	0x0	RW	0x01C4
MacRxPacketStartThresh	MAC_Receive_Packet_Start_Threshold_Register	0xE	RW	0x01C8
MacTxFIFOAlmostEmptyThresh	MAC_Transmit_FIFO_Almost_Empty_Threshold_Register	0x10	RW	0x01CC
MacTxFIFOSpaceAvailableHiThresh	MAC_Transmit_FIFO_Space_Available_Hi_Threshold_Register	0x100	RW	0x01D0
MacTxFIFOSpaceAvailableLoThresh	MAC_Transmit_FIFO_Space_Available_Lo_Threshold_Register	0x10	RW	0x01D4
MacRxFIFOPacketAvailableThresh1	MAC_Receive_FIFO_Packet_Available_Threshold1_Register	0x2010	RW	0x01D8
MacRxFIFOPacketAvailableThresh2	MAC_Receive_FIFO_Packet_Available_Threshold2_Register	0x1	RW	0x01DC
MacStatusAndIRQ	MAC_Status_and_IRQ_Register	0x0	RW	0x01E0
MacIntrEn	MAC_Interrupt_Enable_Register	0x0	RW	0x01E4
MacVLAN_TPID1	MAC_VLAN_TPID1_Register	0x8100	RW	0x01E8
MacVLAN_TPID2	MAC_VLAN_TPID2_Register	0x9100	RW	0x01EC
MacVLAN_TPID3	MAC_VLAN_TPID3_Register	0x88A8	RW	0x01F0
Module1588Ctl	Module_1588_Control_Register	0x0	RW	0x0300
IncrementAttributes	Increment_Attributes_Register	0x0	RW	0x0304
PTPEthertype	PTP_Ethertype_Register	0x0	RW	0x0308
PTPMessageID	PTP_MessageID_Register	0x0	RW	0x030C
PTPUDPPort	PTP_UDP_Port_Register	0x0	RW	0x0310
SystemTimeValueLo	System_Time_Value_Lower_Register	0x0	RO	0x0320
SystemTimeValueUp	System_Time_Value_Upper_Register	0x0	RW	0x0324
SystemTimeAdjustCtlLo	System_Time_Adjust_Control_Lower_Register	0x0	RW	0x0328
SystemTimeAdjustCtlUp	System_Time_Adjust_Control_Upper_Register	0x0	WO	0x032C
TxTimestampValueLo	Transmit_Timestamp_Value_Lower_Register	0x0	RO	0x0330
TxTimestampValueUp	Transmit_Timestamp_Value_Upper_Register	0x0	RO	0x0334
RxTimestampValueLo	Receive_Timestamp_Value_Lower_Register	0x0	RO	0x0340
RxTimestampValueUp	Receive_Timestamp_Value_Upper_Register	0x0	RO	0x0344
RxPTPPacketAttrLo	Receive_PTP_Packet_Attributes_Lower_Register	0x0	RO	0x0348
RxPTPPacketAttrMid	Receive_PTP_Packet_Attributes_Middle_Register	0x0	RO	0x034C
RxPTPPacketAttrUp	Receive_PTP_Packet_Attributes_Upper_Register	0x0	RO	0x0350
Module1588Intr	Module_1588_Interrupt_Register	0x0	RW	0x0360
Module1588IntrEn	Module_1588_Interrupt_Enable_Register	0x0	RW	0x0364

27.3.1 Регистр DMA_Configuration_Register

DMA_Configuration_Register используется для управления основными параметрами контроллера DMA.

Формат регистра DMA_Configuration_Register приведен в Таблица 27.2.

Таблица 27.2. Формат регистра DMA_Configuration_Register

Номер бита	Условное обозначение	Назначение
31:20	-	Резерв
19	Descriptor_Byte_Order_Bit1	Этот бит следует устанавливать только тогда, когда контроллер настроен на работу в 64-битном режиме (установлен бит 18). Данный бит совместно с 13-м битом определяют порядок байтов для дескрипторов. Возможны следующие комбинации {Bit19, Bit13} 2'b00: нормальный формат дескриптора QWORD, как показано в следующих разделах 2'b01: Big Endian в дескрипторе DWORD. 2'b10: DWORDS представлены в формате Big Endian с помощью QWORD. 2'b11: DWORDS и байты в DWORDS имеют формат Big Endian.
18	Mode_64bit	Если бит установлен, то передающий и принимающий DMA выполняют 64-битные транзакции и для дескрипторов и для передачи данных. Внимание! Рекомендуется использовать 64-разрядный режим.
17	Strict_Burst	Если бит установлен, оба DMA работают Strict Burst режиме. В этом режиме DMA ограничивает Burst Size до значения, определенного полем Burst Length или до одиночного DWORD/ QWORD. Если бит сброшен, то оба DMA используют любой Burst Size от 1 до значения, определенного полем Burst Length.
16	Wait_for_Done	Если данный бит установлен, то передающее DMA перед тем как выбрать новый дескриптор пакета ждет готовности FIFO. Если бит снят, то DMA подгружает дескрипторы непрерывно, при условии, что FIFO не переполнен.
15	TX_RX_Arbitration	Выбор сценария арбитража между DMA приема и DMA передачи. Если бит установлен, то используется Round-Robin алгоритм. Если бит снят, то приоритет имеет принимающий DMA (при условии, что TX DMA в данный момент не передает)
14	-	Резерв
13	Descriptor_Byte_Ordering_Bit0	Определяет порядок байтов для дескрипторов В 32-битном режиме работы DMA используется только 13-й бит: 1'b0 – Little Endian ориентация 1'b1 – Big-Endian ориентация В 64-битном режиме добавляется 19-й бит. Кодировка описана ранее.
12:8	Descriptor_Skip_Length	Определяет расстояние между двумя дескрипторами в 32-разрядных словах. Это поле относится как к DMA передачи, так и к DMA приема.
7:6	-	Резерв
5:1	Burst_Length	Отображает максимальное количество 32-разрядных (DWORD) или 64-разрядных (QWORD) слов, передаваемых DMA за одну транзакцию. Возможны следующие значения поля Burst Length: 5'b00001 – 1 DWORD/QWORD 5'b00010 – 2 DWORDS/QWORDS 5'b00100 – 4 DWORDS/QWORDS 5'b01000 – 8 DWORDS/QWORDS 5'b10000 – 16 DWORDS/QWORDS
0	Software_Reset	При установке контроллер DMA сбрасывается в исходное состояние, очищается вся информация о внутреннем состоянии контроллера и буферов. Передающее и принимающее DMA переходят в состояние STOPPED. При сбросе контроллер DMA переходит в нормальный режим работы.

Номер бита	Условное обозначение	Назначение
		Предполагается, что контроллер DMA уже находится в состоянии STOPPED перед установкой Software_reset. Если это не так, то, как только Software_reset деактивирован, контроллер DMA перезапускает себя путем выхода из состояния STOPPED.

27.3.2 Регистр DMA_Control_Register

Регистр управления DMA используется для управления пуском/остановом передающего/принимающего DMA.

Формат регистра DMA_Control_Register приведен в Таблица 27.3.

Таблица 27.3. Формат регистра DMA_Control_Register

Номер бита	Условное обозначение	Назначение
31:2	-	Резерв
1	Start_Stop_Receive_DMA	Когда бит установлен, DMA переходит в состояние RUNNING, GEMAC ядро проверяет список приема текущей позиции для передаваемого кадра. Подгружается новый дескриптор по адресу, указанному в регистре Receive Base Address Register или, если DMA был ранее выключен, восстанавливается значение до остановки. Если дескриптор недоступен, то DMA переходит в состояние SUSPENDED, устанавливается Receive Buffer Unavailable. Команда Start Transmission эффективна только тогда, когда DMA находится в состоянии STOPPED. Если установка данного бита произойдет перед установкой регистра Receive Base Address Register, то поведение GEMAC ядра будет непредсказуемым. Если бит сброшен, то DMA переходит в состояние STOPPED сразу после выполнения текущего приема. В списке приема сохраняется следующая позиция дескриптора и становится текущей при перезапуске DMA. Остановка DMA возможна, если DMA находится в состоянии RUNNING или SUSPENDED.
0	Start_Stop_Transmit_DMA	Когда бит установлен, DMA переходит в состояние RUNNING, GEMAC ядро проверяет список передачи текущей позиции для передаваемого кадра. Подгружается новый дескриптор по адресу, указанному в регистре Transmit Base Address Register или, если DMA был ранее выключен, восстанавливается значение до остановки. Если дескриптор недоступен, то DMA переходит в состояние SUSPENDED, устанавливается Transmit Buffer Unavailable. Команда Start Transmission эффективна только тогда, когда DMA находится в состоянии STOPPED. Если установка данного бита произойдет перед установкой регистра Transmit Base Address Register, то поведение GEMAC ядра будет непредсказуемым. Если бит сброшен, то DMA переходит в состояние STOPPED сразу после выполнения текущей передачи. В списке передачи сохраняется следующая позиция дескриптора и становится текущей при перезапуске DMA. Остановка DMA возможна, если DMA находится в состоянии RUNNING или SUSPENDED.

27.3.3 Регистр DMA_Status_and_IRQ_Register

Регистр статуса и запроса на прерывание DMA отображает текущее состояние DMA и информацию запроса на прерывание относительно различных условий.

Формат регистра DMA_Status_and_IRQ_Register приведен в Таблица 27.4.

Таблица 27.4. Формат регистра DMA_Status_and_IRQ_Register

Номер бита	Условное обозначение	Назначение
31:24	-	Резерв
23:20	Receive_DMA_State	Данные биты отображают текущее состояние принимающего DMA и изменяются динамически. 4'b0000: STOPPED 4'b0001: FETCH_DESCRIPTOR 4'b0010: WAIT_FOR_END_OF_RECEIVE 4'b0011: WAIT_FOR_RXFRAME 4'b0100: SUSPENDED 4'b0101: CLOSE_DESCRIPTOR 4'b0110: FLUSH_BUFFER 4'b0111: PUT_BUFFER 4'b1000: WAIT_FOR_STATUS
19	-	Резерв
18:16	Transmit_DMA_State	Данные биты отображают текущее состояние передающего DMA и изменяются динамически. 3'b000: STOPPED 3'b001: FETCH_DESCRIPTOR 3'b010: Reserved 3'b011: FETCH_DATABUFFER 3'b100: CLOSE_DESCRIPTOR 3'b101: SUSPENDED 3'b110: Reserved 3'b111: Reserved
15:10	-	Резерв
9	Interrupt_1588	Set it indicates that there is an interrupt from the 1588 portion of the GEMAC Core. The Software should read the 1588 Interrupt Register to determine the exact source of 1588 Interrupt. This is cleared only when the bits in the 1588 Interrupt Register are cleared.
8	MAC_Interrupt	Установка данного бита говорит о том, что был установлен запрос на прерывание от MAC части GEMAC ядра. Для уточнения источника запроса на прерывание нужно прочитать регистр MAC Interrupt Register. Данный бит очищается тогда, когда очищены прерывания регистра MAC Interrupt Register.
7	Receive_Missed_Frame_IRQ	Установка данного бита говорит о том, что пакет был пропущен из-за неготовности дескриптора приема. Пакет был сброшен из внутреннего FIFO. Бит устанавливается, если DMA находится в состоянии SUSPENDED, и пришел новый пакет, при этом DMA пытается повторно выбрать дескриптор, а он снова недоступен. IRQ сбрасывается записью единицы.
6	Receive_DMA_Stopped_IRQ	Данный бит устанавливается, когда DMA переходит в состояние STOPPED. IRQ сбрасывается записью единицы.
5	Receive_Descriptor_Unavailable_IRQ	Устанавливается в случае неготовности следующего дескриптора приема. Принимающее DMA переходит в состояние SUSPENDED. Для возобновления процесса подгрузки дескрипторов необходимо

Номер бита	Условное обозначение	Назначение
		обратиться с записью в регистр Receive Poll Demand Register. Если записи регистра Receive Poll Demand Register не было, то дескриптор подгрузится автоматически при приеме нового кадра. IRQ сбрасывается записью единицы.
4	Receive_Transfer_Done_IRQ	Установка бита означает успешный прием и передачу в память пакета, а также запись статуса пакета в поле RDES0 последнего дескриптора. Принимающий DMA остается в состоянии RUNNING и выбирает следующий дескриптор. В режиме Receive Interrupt Mitigation Mode запрос на прерывание выставляется, когда в память передано запрограммированное количество пакетов, либо при срабатывании счетчика Receive Interrupt Timeout, отсчитывающего время с момента передачи в память первого пакета. IRQ сбрасывается записью единицы.
3	-	Резерв
2	Transmit_DMA_Stopped_IRQ	Данный бит устанавливается, когда DMA переходит в состояние STOPPED. IRQ сбрасывается записью единицы.
1	Transmit_Descriptor_Unavailable_IRQ	Устанавливается в случае неготовности следующего дескриптора передачи. Передающее DMA переходит в состояние SUSPENDED. Если не включен Transmit Auto Polling, то для возобновления процесса подгрузки дескрипторов необходимо обратиться с записью в регистр Transmit Poll Demand Register. IRQ сбрасывается записью единицы.
0	Transmit_Transfer_Done_IRQ	Установка этого бита означает, что передача пакета выполнена успешно, и Interrupt On Completion (TDES[31]) установлен в первом дескрипторе пакета. Если бит Wait For Done не установлен, то запрос на прерывание устанавливается, когда пакет передан в Transmit FIFO. Если бит Wait For Done установлен, то запрос на прерывание устанавливается, когда пакет передан на RGMII интерфейс. IRQ сбрасывается записью единицы.

27.3.4 Регистр DMA_Interrupt_Enable_Register

Регистр маски запросов на прерывание.

Формат регистра DMA_Interrupt_Enable_Register приведен в Таблица 27.5.

Таблица 27.5. Формат регистра DMA_Interrupt_Enable_Register

Номер бита	Условное обозначение	Назначение
31:10	-	Резерв
9	Interrupt_1588_Enable	Маска прерывания 1588 1- разрешено 0 – запрещено
8	MAC_Interrupt_Enable	Маска прерывания MAC Interrupt 1- разрешено 0 – запрещено
7	Receive_Missed_Frame_Interrupt_Enable	Маска прерывания Receive Missed Frame IRQ 1- разрешено 0 – запрещено
6	Receive_DMA_Stopped_Interrupt_Enable	Маска прерывания Receive DMA Stopped IRQ 1- разрешено 0 – запрещено

Номер бита	Условное обозначение	Назначение
5	Receive_Descriptor_Unavailable_Interrupt_Enable	Маска прерывания Receive Descriptor Unavailable IRQ 1- разрешено 0 – запрещено
4	Receive_Transfer_Done_Interrupt_Enable	Маска прерывания Receive Transfer Done IRQ 1- разрешено 0 – запрещено
3	-	Резерв
2	Transmit_DMA_Stopped_Interrupt_Enable	Маска прерывания Transmit DMA Stopped IRQ 1- разрешено 0 – запрещено
1	Transmit_Descriptor_Unavailable_Interrupt_Enable	Маска прерывания Transmit Descriptor Unavailable IRQ 1- разрешено 0 – запрещено
0	Transmit_Transfer_Done_Interrupt_Enable	Маска прерывания Transmit Transfer Done IRQ 1- разрешено 0 – запрещено

27.3.5 Регистр DMA_Transmit_Auto_Poll_Counter_Register

Регистр счетчика автоматического опроса передающего DMA используется для определения частоты опроса, если Transmit DMA находится в состоянии ожидания. Когда Transmit DMA находится в состоянии ожидания, выборка дескриптора повторяется в соответствии с периодом, записанным в счетчик, или когда осуществляется запись нового значения счетчика.

Формат регистра R DMA_Transmit_Auto_Poll_Counter_Register EG приведен в Таблица 27.6.

Таблица 27.6. Формат регистра DMA_Transmit_Auto_Poll_Counter_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:0	Transmit_Auto_Poll_Value	Значение, определяющее количество тактов ожидания частоты AXI в состоянии SUSPENDED перед повторной выборкой дескриптора. Если дескриптор недоступен, то DMA возвращается в состояние SUSPENDED и устанавливает Transmit Descriptor Unavailable Interrupt. Внутренний счетчик Auto Poll Counter используется только тогда, когда Transmit DMA находится в состоянии SUSPENDED и сбрасывается в противном случае. Функция автоматического опроса отключается, если значение, записанное в регистр, равно нулю.

27.3.6 Регистр DMA_Transmit_Poll_Demand_Register

Регистр запуска выборки дескриптора передающего DMA используется для указания Transmit DMA о необходимости загрузки нового дескриптора, при условии, что DMA находится в состоянии ожидания. Здесь регистрируется сам факт записи, а записываемое значение игнорируется.

Формат регистра DMA_Transmit_Poll_Demand_Register приведен в Таблица 27.7.

Таблица 27.7. Формат регистра DMA_Transmit_Poll_Demand_Register

Номер бита	Условное обозначение	Назначение
31:0	Transmit_Poll_Demand	Если Transmit DMA находится в состоянии SUSPENDED, и осуществляется запись в данный регистр любого значения, то ядро GEMAC осуществляет выборку нового дескриптора передачи. Если дескриптор недоступен, то DMA возвращается в состояние SUSPENDED и устанавливает Transmit Descriptor Unavailable Interrupt.

27.3.7 Регистр DMA_Receive_Poll_Demand_Register

Регистр запуска выборки дескриптора Receive DMA используется для указания Receive DMA о необходимости загрузки нового дескриптора, при условии, что DMA находится в состоянии ожидания. Здесь регистрируется сам факт записи, а записываемое значение игнорируется.

Формат регистра DMA_Receive_Poll_Demand_Register приведен в Таблица 27.8.

Таблица 27.8. Формат регистра DMA_Receive_Poll_Demand_Register

Номер бита	Условное обозначение	Назначение
31:0	Receive_Poll_Demand	Если Receive DMA находится в состоянии SUSPENDED, и осуществляется запись в данный регистр любого значения, то ядро GEMAC осуществляет выборку нового дескриптора приема. Если дескриптор недоступен, то DMA возвращается в состояние SUSPENDED и устанавливает Receive Descriptor Unavailable Interrupt.

27.3.8 Регистр DMA_Transmit_Base_Address_Register

Регистр базового адреса Transmit DMA используется, чтобы указать на начало списка дескрипторов передачи в адресном пространстве памяти.

Формат регистра DMA_Transmit_Base_Address_Register приведен в Таблица 27.9.

Таблица 27.9. Формат регистра DMA_Transmit_Base_Address_Register

Номер бита	Условное обозначение	Назначение
31:0	Transmit_Base_Address	Данный регистр содержит стартовый адрес списка дескрипторов передачи во внешней памяти. При пуске Transmit DMA, значение регистра используется для выборки дескриптора (только если регистр обновлен), в противном случае используется адрес, сохраненный перед остановкой DMA. В 32-разрядном режиме все дескрипторы и значение в этом регистре должны быть выровнены по 32-битной границе В 64-разрядном режиме все дескрипторы и значение в этом регистре должны быть выровнены по 64-битной границе Запись регистра необходимо осуществлять только в момент нахождения DMA в состоянии STOPPED.

27.3.9 Регистр DMA_Receive_Base_Address_Register

Регистр базового адреса Receive DMA используется, чтобы указать на начало списка дескрипторов приема в адресном пространстве памяти.

Формат регистра DMA_Receive_Base_Address_Register приведен в Таблица 27.10.

Таблица 27.10. Формат регистра DMA_Receive_Base_Address_Register

Номер бита	Условное обозначение	Назначение
31:0	Receive_Base_Address	Данный регистр содержит стартовый адрес списка дескрипторов приема во внешней памяти. При пуске Receive DMA значение регистра используется для выборки дескриптора (только если регистр обновлен), в противном случае используется адрес, сохраненный перед остановкой DMA. В 32-разрядном режиме все дескрипторы и значение в этом регистре должны быть выровнены по 32-битной границе В 64-разрядном режиме все дескрипторы и значение в этом регистре должны быть выровнены по 64-битной границе Запись регистра необходимо осуществлять только в момент нахождения DMA в состоянии STOPPED.

27.3.10 Регистр DMA_Missed_Frame_Counter_Register

Регистр счетчика пропущенных кадров отображает количество кадров, которые были пропущены из-за неготовности дескриптора приема. Если Receive DMA приостановлен, и был принят новый кадр, то DMA пытается подгрузить дескриптор приема. Если дескриптор недоступен, то фрейм сбрасывается из Receive FIFO. Этот счетчик отображает общее количество кадров, которые были пропущены с момента предыдущего чтения данного регистра.

Формат регистра DMA_Missed_Frame_Counter_Register приведен в Таблица 27.11.

Таблица 27.11. Формат регистра DMA_Missed_Frame_Counter_Register

Номер бита	Условное обозначение	Назначение
31	Missed_Frame_Counter_Overflow_IRQ	Бит устанавливается при переполнении Missed Frame Counter. Установка этого бита ведет к формированию запроса на прерывание. Бит сбрасывается, если регистр прочитан.
30:0	Missed_Frame_Counter	Missed Frame Counter отображает число пропущенных пакетов из-за неготовности дескриптора приема. Счетчик отображает общее число пропущенных пакетов с момента последнего чтения данного регистра. Счетчик сбрасывается, если регистр прочитан.

27.3.11 Регистр DMA_Stop_Flush_Counter_Register

Регистр счетчика сброшенных кадров при остановленном DMA отображает количество сброшенных кадров, которые были сброшены из-за того, что DMA находится в состоянии останова. Этот счетчик отображает общее количество кадров, которые были пропущены с момента предыдущего чтения данного регистра. Когда счетчик переполняется, формируется запрос на прерывание.

Формат регистра DMA_Stop_Flush_Counter_Register приведен в Таблица 27.12.

Таблица 27.12. Формат регистра DMA_Stop_Flush_Counter_Register

Номер бита	Условное обозначение	Назначение
31	Stop_Counter_Overflow_IRQ	Бит устанавливается при переполнении Stop Flush Counter. Установка этого бита ведет к формированию запроса на прерывание. Бит сбрасывается, если регистр прочитан.
30:0	Stop_Flush_Counter	Stop Flush Counter отображает число пропущенных пакетов из-за того, что DMA находится в состоянии STOPPED. Счетчик отображает общее число пропущенных пакетов с момента последнего чтения данного регистра. Счетчик сбрасывается, если регистр прочитан.

27.3.12 Регистр DMA_Receive_Interrupt_Mitigation_Control

Данный регистр позволяет управлять поведением запросом на прерывание, возникающим во время передачи в память принятого пакета.

Формат регистра DMA_Receive_Interrupt_Mitigation_Control приведен в Таблица 27.13.

Таблица 27.13. Формат регистра DMA_Receive_Interrupt_Mitigation_Control

Номер бита	Условное обозначение	Назначение
31	Receive_Transfer_Done_Interrupt_Mitigation_Control_Enable	Установка данного бита включает режим Mitigation Mode с использованием счетчиков пакетов/времени. Если бит снят, то запрос на прерывание Receive Transfer Done Interrupt формируется после каждого принятого и переданного в память пакета.
30	Receive_Interrupt_Frame_Counter_Mode	Если осуществляется работа в режиме Mitigation Mode, и установлен данный бит, то после формирования запроса на прерывание Receive Transfer Done Interrupt значение поля Receive Interrupt Frame Counter сбрасывается в 8'h01. Если данный бит не установлен, то поле Receive Interrupt Frame Counter после формирования запроса на прерывание Receive Transfer Done Interrupt сохраняет ранее запрограммированное значение.
29:28	-	Резерв
27:8	Receive_Interrupt_Timeout_Counter	Отображает максимальное время в тактах частоты ahb_clk между первым принятым пакетом и запросом на прерывание Receive Transfer Done Interrupt. Если включен режим Interrupt Mitigation, то Receive Transfer Done Interrupt выставляется после приема N пакетов, где N – определяется полем Receive Interrupt Frame Counter. Запрос на прерывание Receive Transfer Done Interrupt выставляется также, если счетчик Timeout Counter (начинающий отсчет после приема первого пакета) достиг значения Receive Interrupt Timeout Counter раньше, чем было принято N пакетов.
7:0	Receive_Interrupt_Frame_Counter	Отображает количество кадров, которые необходимо принять для формирования прерывания Receive Transfer Done Interrupt. Данное значение достоверно, если установлен 31-й бит. Можно запрограммировать значение от 1 до 255

27.3.13 Регистр DMA_Current_Transmit_Descriptor_Pointer_Register

Регистр указателя текущего дескриптора Transmit DMA содержит адрес дескриптора, используемого в данный момент. Данный регистр доступен только по чтению.

Формат регистра DMA_Current_Transmit_Descriptor_Pointer_Register приведен в Таблица 27.14.

Таблица 27.14. Формат регистра DMA_Current_Transmit_Descriptor_Pointer_Register

Номер бита	Условное обозначение	Назначение
31:0	Current_Transmit_ Descriptor_Pointer	Данное поле содержит указатель на текущий дескриптор Transmit DMA. Указатель выровнен по 32-разрядной или 64-разрядной границе в зависимости от режима работы.

27.3.14 Регистр DMA_Current_Transmit_Buffer_Pointer_Register

Регистр указателя текущего буфера Transmit DMA содержит адрес буфера, используемого в данный момент. Данный регистр доступен только по чтению.

Формат регистра DMA_Current_Transmit_Buffer_Pointer_Register приведен в Таблица 27.15.

Таблица 27.15. Формат регистра DMA_Current_Transmit_Buffer_Pointer_Register

Номер бита	Условное обозначение	Назначение
31:0	Current_Transmit_ Buffer_Pointer	Данное поле содержит указатель на текущий буфер Transmit DMA.

27.3.15 Регистр DMA_Current_Receive_Descriptor_Pointer_Register

Регистр указателя текущего дескриптора Receive DMA содержит адрес дескриптора, используемого в данный момент. Данный регистр доступен только по чтению.

Формат регистра DMA_Current_Receive_Descriptor_Pointer_Register приведен в Таблица 27.16.

Таблица 27.16. Формат регистра DMA_Current_Receive_Descriptor_Pointer_Register

Номер бита	Условное обозначение	Назначение
31:0	Current_Receive_D escriptor_Pointer	Данное поле содержит указатель на текущий дескриптор Receive DMA. Указатель выровнен по 32-разрядной или 64-разрядной границе в зависимости от режима работы.

27.3.16 Регистр DMA_Current_Receive_Buffer_Pointer_Register

Регистр указателя текущего буфера Receive DMA содержит адрес буфера используемого в данный момент. Данный регистр доступен только по чтению.

Формат регистра DMA_Current_Receive_Buffer_Pointer_Register приведен в Таблица 27.17.

Таблица 27.17. Формат регистра DMA_Current_Receive_Buffer_Pointer_Register

Номер бита	Условное обозначение	Назначение
31:0	Current_Receive_Buffer_Pointer	Данное поле содержит указатель на текущий буфер Receive DMA.

27.3.17 Регистр MAC Global Control Register

Глобальный регистр управления используется для программирования глобальных параметров MAC контроллера GEMAC.

Формат регистра MAC Global Control Registerприведен в Таблица 27.18.

Таблица 27.18. Формат регистра MAC Global Control Register

Номер бита	Условное обозначение	Назначение
31:10	-	Резерв
9	Magic_Packet_Wakeup_Mode	Если установлено, ядро GEMAC работает в режиме пробуждения (Wakeup Mode) и блокирует принятые кадры для записи в RXFIFO Mac. Событие Wakeup генерируется, когда принят Magic Packet, поле DA которого проходит фильтрацию адресов и имеет подпись Magic Packet в содержимом пакета.
8	Unicast_Wakeup_Mode	Если установлено, ядро GEMAC работает в режиме пробуждения (Wakeup Mode) и блокирует принятые кадры для записи в RXFIFO Mac. Событие Wakeup генерируется, когда принят пакет с уникальным адресом, поле DA которого соответствует значению, запрограммированному в регистре MAC_Address0.
7:5	-	Резерв
4	Reset_Tx_Stat_Counters	Установка данного бита инициализирует сброс всех статистических счетчиков тракта передачи. Процесс инициализации начинается после того, как бит очищен.
3	Reset_Rx_Stat_Counters	Установка данного бита инициализирует сброс всех статистических счетчиков тракта приема. Процесс инициализации начинается после того, как бит очищен.
2	Duplex_Mode	Если бит установлен, то ядро работает в Full-Duplex режиме, в котором возможен одновременный прием и передача. Если бит сброшен, то ядро работает в Half Duplex режиме и, в соответствии с CSMA/CD протоколом, отслеживает коллизии и осуществляет Back-Off столкновения. Изменение режима возможно только при выключенном приемнике и передатчике. 1’b0: Half-Duplex Mode 1’b1: Full-Duplex Mode При работе на скорости 1000Mbps поддерживается только Full-Duplex Mode
1:0	Speed	Определяет скорость Ethernet интерфейса GEMAC ядра. Выбранной скорости должны соответствовать частоты txclk и rxclk. 2’b00: 10 Mbps 2’b01: 100 Mbps 2’b10: 1000 Mbps 2’b11: Reserved

27.3.18 Регистр MAC Transmit Control Register

Регистр управления передачей используется для установки параметров передающей части MAC в GEMAC ядре.

Формат регистра MAC Transmit Control Register приведен в Таблица 27.19.

Таблица 27.19. Формат регистра MAC Transmit Control Register

Номер бита	Условное обозначение	Назначение
31:10	-	Резерв
9:7	Preamble_Length	Определяет количество байт преамбулы передаваемого кадра. По спецификации преамбула должна иметь 7 байт. 3'b000: 7 байт 3'b001: 1 байт 3'b010: 2 байта 3'b011: 3 байта 3'b100: 4 байта 3'b101: 5 байт 3'b110: 6 байт 3'b111: 7 байт
6:4	IFG_Length	Определяет минимальный IPG (Inter Packet/Frame Gap) вставляемый между исходящими пакетами. По спецификации минимальный IPG между пакетами должен состоять из 96 bit-times. 3'b000: 96 Bit Times 3'b001: 64 Bit Times 3'b010: 128 Bit Times 3'b011: 256 Bit Times 3'b100: 24 Bit Times 3'b101: 32 Bit Times 3'b110: 40 Bit Times 3'b111: 48 Bit Times При установке значения IPG менее 96 bit-times могут возникнуть проблемы с совместимостью с другими сетевыми устройствами. Установка значения IPG более 96 bit-times ведет к ухудшению производительности.
3	Transmit_Auto_Retry	Если бит установлен, то при работе в режиме Half-Duplex и возникновении коллизии контроллер автоматически повторяет передачу пакета. Для правильной работы в полудуплексном режиме этот бит должен быть установлен в «1».
2	Disable_FCS_Insertion	Если бит установлен, то вычисление и вставка FCS не производится. Вставка FCS поля отключается для всех пакетов. Если вставка FCS отключена, то предполагается, что кадр, сформированный в памяти, уже содержит поле FCS и соответствует требованию MinFrameSize.
1	Invert_FCS	Если бит установлен, то GEMAC ядро инвертирует поле FCS исходящего пакета. Если бит сброшен, то вставляется нормальное поле FCS.
0	Transmit_Enable	При установке данного бита включается передатчик ядра и передает пакеты из Transmit FIFO на MII/GMII интерфейс. Если бит сброшен – передатчик отключается, кадры не передаются.

27.3.19 Регистр MAC Receive Control Register

Регистр управления приемом MAC используется для установки параметров приемной части MAC в GEMAC ядре.

Формат регистра MAC Receive Control Register приведен в Таблица 27.20.

Таблица 27.20. Формат регистра MAC Receive Control Register

Номер бита	Условное обозначение	Назначение
31:7	-	Резерв
6	Account_VLAN	Согласно спецификации, значение minFrameSize и maxFrameSize непомеченных пакетов равны 64 и 1518 байтам соответственно. Значение размера помеченного (VLAN Tagged) пакета увеличено на 4 байта VLAN Tag и значение maxFrameSize увеличено до 1522 байт. GEMAC ядро поддерживает до 3 VLAN Tags на пакет. В зависимости от количества VLAN Tag во входящем пакете поле maxFrameSize может быть расширено для поддержки до трех VLAN Tag (12 байт). Если данный бит установлен, то поле maxFrameSize может быть расширено до 3 VLAN Tags (12 бит), в зависимости от количества VLAN полей в принимаемом кадре. Если бит сброшен, то значение, запрограммированное в регистре Maximum Frame Size Register не изменяется.
5	Pass_Bad_Frames	При работе в режиме store-and-forward пакеты, принятые с ошибками, могут быть сброшены из Receive FIFO без передачи в память. Если данный бит установлен, то пакеты, принятые с ошибкой, передаются в память с отметкой ошибки в поле статуса (TDES0 последнего дескриптора). Если бит сброшен, пакеты с ошибками не передаются в память, а сбрасываются из Receive FIFO.
4	Status_First	При работе в режиме store-and-forward пакет может быть передан с полем статуса как первые данные (перед SOP), или как последние данные (после EOP). Если данный бит установлен, то поле статуса передается перед SOP, если бит сброшен – после EOP. Для правильной работы контроллера данный бит должен быть всегда сброшен.
3	Store_and_Forward	Если бит установлен, то контроллер работает в режиме store-and-forward. Пакет передается в память только тогда, когда он полностью загрузится в Receive FIFO. Пакеты с ошибками могут быть сброшены из Receive FIFO без передачи в память. Если бит сброшен, то контроллер работает в режиме cut-through. Пакеты начинают передаваться в память без ожидания завершения их приема. При работе в режиме store-and-forward размер FIFO определяет максимальный размер принимаемого пакета.
2	Strip_FCS	Если бит установлен, то у кадра, при передаче в память, удаляется поле FCS и корректируется его длина. Если бит сброшен, то FCS не удаляется, а передается вместе с кадром в память. При этом проверка FCS проводится (если она не отключена) для каждого кадра, также сохраняется и результат проверки.
1	Disable_FCS_Checking	Если бит установлен, то проверка поля FCS входящего кадра не проводится, и CRC Error Status для всех кадров не устанавливается. Если бит сброшен, то контроллер функционирует в обычном режиме, проверяя FCS для каждого входящего кадра. Кадры с ошибкой CRC помечаются или сбрасываются.
0	Receive_Enable	При установке данного бита включается приемник ядра и передает пакеты с MII/GMII интерфейс в Receive FIFO. Если бит сброшен – приемник отключается, кадры не принимаются.

27.3.20 Регистр MAC Maximum Frame Size Register

Регистр максимального размера кадра MAC используется для установки значение поля MaxFrameSize, по которому проверяется MaxFrameLength нарушения.

Формат регистра MAC Maximum Frame Size Register приведен в Таблица 27.21.

Таблица 27.21. Формат регистра MAC Maximum Frame Size Register

Номер бита	Условное обозначение	Назначение
31:14	-	Резерв
13:0	Maximum_Frame_Size	Данное поле определяет Maximum Frame Size непомеченного пакета и используется для проверки ограничения MaxFrameLength. В режиме Store-and-Forward любой пакет, больший, чем это значение, сбрасывается из Receive FIFO, если не установлен бит Pass Bad Frames. Если Pass Bad Frames установлен, то пакет с длиной, большей значения MaxFrameLength передается в память с установкой бита ошибки в статусе пакета (в TDES0 последнего дескриптора).

27.3.21 Регистр MAC Transmit Jabber Size Register

Регистр размера Jabber данных передатчика MAC используется для установки размера Jabber данных передаваемого кадра. Когда длина передающегося кадра превышает это значение, кадр усекается и выдается EOP-ERROR.

Формат регистра MAC Transmit Jabber Size Register приведен в Таблица 27.22.

Таблица 27.22. Формат регистра MAC Transmit Jabber Size Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:0	Transmit_Jabber_Size	Данное поле определяет Jabber Size передаваемого пакета. Если длина исходящего пакета превышает значение, запрограммированное в этом поле, то пакет признается бессмысленным, и его передача прекращается с выставлением EOP-ERROR.

27.3.22 Регистр MAC Receive Jabber Size Register

Регистр размера Jabber данных приемника MAC используется для установки размера Jabber данных принимаемого кадра. Когда длина принимаемого кадра превышает это значение, тогда он обрезается и выставляется ошибка в регистре статуса «Jabber Error». Остаток принимаемых данных игнорируется.

Формат регистра MAC Receive Jabber Size Register приведен в Таблица 27.23.

Таблица 27.23. Формат регистра MAC Receive Jabber Size Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв

Номер бита	Условное обозначение	Назначение
15:0	Receive_Jabber_Size	Данное поле определяет Jabber Size принимаемого пакета. Если длина входящего пакета превышает значение, запрограммированное в этом поле, то пакет признается бессмысленным, и его прием прекращается. В статусе кадра отмечается Jabber Error. В режиме Store-and-Forward любой пакет сбрасывается из Receive FIFO, если не установлен бит Pass Bad Frames.

27.3.23 Регистр MAC Address Control Register

Регистр управления MAC адресом используется для управления различными аспектами контроля MAC адресов, который выполняется для всех входящих фреймов. GEMAC ядро выполняет фильтрацию адреса в соответствии со значением Destination Address для всех входящих фреймов. Для фильтрации Unicast адресов используется четыре регистра MAC адреса, для фильтрации Multicast фреймов используется Multicast Хеш-таблица. В нормальном режиме работы, в память передаются лишь те кадры, которые прошли адресную фильтрацию. Кадры, которые не прошли адресную фильтрацию, сбрасываются в Receive FIFO. Для каждого фрейма результат адресной фильтрации отображается в статусе пакета, который записан в поле TDES0 последнего дескриптора. При работе в режиме Promiscuous, принимаются и передаются в память все фреймы, независимо от адресной фильтрации.

Формат регистра MAC Address Control Register приведен в Таблица 27.24.

Таблица 27.24. Формат регистра MAC Address Control Register

Номер бита	Условное обозначение	Назначение
31:9	-	Резерв
8	Promiscuous_Mode	Если данный бит установлен, то принимаются и передаются в память все пакеты независимо от значения поля Destination Address. В поле статуса пакета отмечается результат адресной фильтрации.
7	Inverse_MAC_Address4_Enable	Если бит установлен, то выполняется инверсная фильтрация поля Destination Address со значением регистра MAC Address Register#4. Инверсная фильтрация включается только при условии, что установлен бит MAC Address#4 Enable. Если сброшен, то производится нормальная фильтрация.
6	Inverse_MAC_Address3_Enable	Если бит установлен, то выполняется инверсная фильтрация поля Destination Address со значением регистра MAC Address Register#3. Инверсная фильтрация включается только при условии, что установлен бит MAC Address#3 Enable. Если сброшен, то производится нормальная фильтрация.
5	Inverse_MAC_Address2_Enable	Если бит установлен, то выполняется инверсная фильтрация поля Destination Address со значением регистра MAC Address Register#2. Инверсная фильтрация включается только при условии, что установлен бит MAC Address#2 Enable. Если сброшен, то производится нормальная фильтрация.
4	Inverse_MAC_Address1_Enable	Если бит установлен, то выполняется инверсная фильтрация поля Destination Address со значением регистра MAC Address Register#1. Инверсная фильтрация включается только при условии, что

Номер бита	Условное обозначение	Назначение
		установлен бит MAC Address#1 Enable. Если сброшен, то производится нормальная фильтрация.
3	MAC_Address4_Enable	Если бит установлен, то MAC Address Register#4 используется для выполнения адресной фильтрации поля Destination Address всех входящих пакетов. Если бит сброшен, то MAC Address Register#4 не используется для выполнения адресной фильтрации.
2	MAC_Address3_Enable	Если бит установлен, то MAC Address Register#3 используется для выполнения адресной фильтрации поля Destination Address всех входящих пакетов. Если бит сброшен, то MAC Address Register#3 не используется для выполнения адресной фильтрации.
1	MAC_Address2_Enable	Если бит установлен, то MAC Address Register#2 используется для выполнения адресной фильтрации поля Destination Address всех входящих пакетов. Если бит сброшен, то MAC Address Register#2 не используется для выполнения адресной фильтрации.
0	MAC_Address1_Enable	Если бит установлен, то MAC Address Register#1 используется для выполнения адресной фильтрации поля Destination Address всех входящих пакетов. Если бит сброшен, то MAC Address Register#1 не используется для выполнения адресной фильтрации.

27.3.24 Регистр MAC MDIO Clock Division Control Register

Регистр управления частотой MDIO используется для того, чтобы сконфигурировать соотношение частоты MDC с системной частотой ahb_clk.

Формат регистра MAC MDIO Clock Division Control Register приведен в Таблица 27.25.

Таблица 27.25. Формат регистра MAC MDIO Clock Division Control Register

Номер бита	Условное обозначение	Назначение
31:8	-	Резерв
7:0	MDC_Clock_Division_Control	Определяет соотношение между частотами MDC Clock и ahb_clk. По умолчанию стоит значение 64, т.е. на MDC Clock 32 периода частоты ahb_clk держится «1», затем 32 периода частоты ahb_clk держится «0». Поддерживаются только четные значения до 8’hFE.

27.3.25 Регистр MAC_Address1_High_Register

Регистр старшей части MAC адреса#1 содержит первые два байта MAC адреса#1, как представлено в каноническом формате. Канонический формат показывает, в какой последовательности байты передаются на интерфейс Ethernet. Регистры средней части MAC адреса#1 и младшей части MAC адреса#1 содержат остаток 48-битного MAC адреса#1.

Формат регистра MAC_Address1_High_Register приведен в Таблица 27.26.

Таблица 27.26. Формат регистра MAC_Address1_High_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:8	MAC_Address1_Second_Byte	В данном поле содержится второй байт MAC Address. Соответствует второму передаваемому байту поля DA

Номер бита	Условное обозначение	Назначение
7:0	MAC_Address1_First_Byte	В данном поле содержится первый байт MAC Address. Соответствует первому передаваемому байту поля DA

27.3.26 Регистр MAC_Address1_Med_Register

Регистр средней части MAC адреса#1 содержит средние два байта MAC адреса#1, как представлено в каноническом формате. Канонический формат показывает, в какой последовательности байты передаются на интерфейс Ethernet. Регистры старшей части MAC адреса#1 и младшей части MAC адреса#1 содержат остаток 48-битного MAC адреса#1.

Формат регистра R MAC_Address1_Med_Register EG приведен в Таблица 27.27.

Таблица 27.27. Формат регистра MAC_Address1_Med_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:8	MAC_Address1_Fourth_Byte	В данном поле содержится четвертый байт MAC Address. Соответствует четвертому передаваемому байту поля DA
7:0	MAC_Address1_Third_Byte	В данном поле содержится третий байт MAC Address. Соответствует первому передаваемому байту поля DA

27.3.27 Регистр MAC_Address1_Low_Register

Регистр младшей части MAC адреса#1 содержит младшие два байта MAC адреса#1, как представлено в каноническом формате. Канонический формат показывает, в какой последовательности байты передаются на интерфейс Ethernet. Регистры старшей части MAC адреса#1 и средней части MAC адреса#1 содержат остаток 48-битного MAC адреса#1.

Формат регистра MAC_Address1_Low_Register приведен в Таблица 27.28.

Таблица 27.28. Формат регистра MAC_Address1_Low_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:8	MAC_Address1_Sixth_Byte	В данном поле содержится шестой байт MAC Address. Соответствует шестому передаваемому байту поля DA
7:0	MAC_Address1_Fifth_Byte	В данном поле содержится пятый байт MAC Address. Соответствует пятому передаваемому байту поля DA

27.3.28 Регистр MAC_Address2_High_Register

Регистр старшей части MAC адреса#2 содержит первые два байта MAC адреса#2, как представлено в каноническом формате. Канонический формат показывает, в какой последовательности байты передаются на интерфейс Ethernet. Регистры средней части MAC адреса#2 и младшей части MAC адреса#2 содержат остаток 48-битного MAC адреса#2.

Формат регистра MAC_Address2_High_Register приведен в Таблица 27.29.

Таблица 27.29. Формат регистра MAC_Address2_High_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:8	MAC_Address2_Second_Byte	В данном поле содержится второй байт MAC Address. Соответствует второму передаваемому байту поля DA
7:0	MAC_Address2_First_Byte	В данном поле содержится первый байт MAC Address. Соответствует первому передаваемому байту поля DA

27.3.29 Регистр MAC_Address2_Med_Register

Регистр средней части MAC адреса#2 содержит средние два байта MAC адреса#2, как представлено в каноническом формате. Канонический формат показывает, в какой последовательности байты передаются на интерфейс Ethernet. Регистры старшей части MAC адреса#2 и младшей части MAC адреса#2 содержат остаток 48-битного MAC адреса#2.

Формат регистра R MAC_Address2_Med_Register EG приведен в Таблица 27.30.

Таблица 27.30. Формат регистра MAC_Address2_Med_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:8	MAC_Address2_Fourth_Byte	В данном поле содержится четвертый байт MAC Address. Соответствует четвертому передаваемому байту поля DA
7:0	MAC_Address2_Third_Byte	В данном поле содержится третий байт MAC Address. Соответствует первому передаваемому байту поля DA

27.3.30 Регистр MAC_Address2_Low_Register

Регистр младшей части MAC адреса#2 содержит младшие два байта MAC адреса#2, как представлено в каноническом формате. Канонический формат показывает, в какой последовательности байты передаются на интерфейс Ethernet. Регистры старшей части MAC адреса#2 и средней части MAC адреса#2 содержат остаток 48-битного MAC адреса#2.

Формат регистра MAC_Address2_Low_Register приведен в Таблица 27.31.

Таблица 27.31. Формат регистра MAC_Address2_Low_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:8	MAC_Address2_Sixth_Byte	В данном поле содержится шестой байт MAC Address. Соответствует шестому передаваемому байту поля DA
7:0	MAC_Address2_Fifth_Byte	В данном поле содержится пятый байт MAC Address. Соответствует пятому передаваемому байту поля DA

27.3.31 Регистр MAC_Address3_High_Register

Регистр старшей части MAC адреса#3 содержит первые два байта MAC адреса#3, как представлено в каноническом формате. Канонический формат показывает, в какой последовательности байты передаются на интерфейс Ethernet. Регистры средней части MAC адреса#3 и младшей части MAC адреса#3 содержат остаток 48-битного MAC адреса#3.

Формат регистра MAC_Address3_High_Register приведен в Таблица 27.32.

Таблица 27.32. Формат регистра MAC_Address3_High_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:8	MAC_Address3_Second_Byte	В данном поле содержится второй байт MAC Address. Соответствует второму передаваемому байту поля DA
7:0	MAC_Address3_First_Byte	В данном поле содержится первый байт MAC Address. Соответствует первому передаваемому байту поля DA

27.3.32 Регистр MAC_Address3_Med_Register

Регистр средней части MAC адреса#3 содержит средние два байта MAC адреса#3, как представлено в каноническом формате. Канонический формат показывает, в какой последовательности байты передаются на интерфейс Ethernet. Регистры старшей части MAC адреса#3 и младшей части MAC адреса#3 содержат остаток 48-битного MAC адреса#3.

Формат регистра R MAC_Address3_Med_Register EG приведен в Таблица 27.33.

Таблица 27.33. Формат регистра MAC_Address3_Med_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:8	MAC_Address3_Fourth_Byte	В данном поле содержится четвертый байт MAC Address. Соответствует четвертому передаваемому байту поля DA
7:0	MAC_Address3_Third_Byte	В данном поле содержится третий байт MAC Address. Соответствует первому передаваемому байту поля DA

27.3.33 Регистр MAC_Address3_Low_Register

Регистр младшей части MAC адреса#3 содержит младшие два байта MAC адреса#3, как представлено в каноническом формате. Канонический формат показывает, в какой последовательности байты передаются на интерфейс Ethernet. Регистры старшей части MAC адреса#3 и средней части MAC адреса#3 содержат остаток 48-битного MAC адреса#3.

Формат регистра MAC_Address3_Low_Register приведен в Таблица 27.34.

Таблица 27.34. Формат регистра MAC_Address3_Low_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:8	MAC_Address3_Sixth_Byte	В данном поле содержится шестой байт MAC Address. Соответствует шестому передаваемому байту поля DA
7:0	MAC_Address3_Fifth_Byte	В данном поле содержится пятый байт MAC Address. Соответствует пятому передаваемому байту поля DA

27.3.34 Регистр MAC_Address4_High_Register

Регистр старшей части MAC адреса#4 содержит первые два байта MAC адреса#4, как представлено в каноническом формате. Канонический формат показывает, в какой последовательности байты передаются на интерфейс Ethernet. Регистры средней части MAC адреса#4 и младшей части MAC адреса#4 содержат остаток 48-битного MAC адреса#4.

Формат регистра MAC_Address4_High_Register приведен в Таблица 27.35.

Таблица 27.35. Формат регистра MAC_Address4_High_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:8	MAC_Address4_Second_Byte	В данном поле содержится второй байт MAC Address. Соответствует второму передаваемому байту поля DA
7:0	MAC_Address4_First_Byte	В данном поле содержится первый байт MAC Address. Соответствует первому передаваемому байту поля DA

27.3.35 Регистр MAC_Address4_Med_Register

Регистр средней части MAC адреса#4 содержит средние два байта MAC адреса#4, как представлено в каноническом формате. Канонический формат показывает, в какой последовательности байты передаются на интерфейс Ethernet. Регистры старшей части MAC адреса#4 и младшей части MAC адреса#4 содержат остаток 48-битного MAC адреса#4.

Формат регистра R MAC_Address4_Med_Register EG приведен в Таблица 27.36.

Таблица 27.36. Формат регистра MAC_Address4_Med_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:8	MAC_Address4_Fourth_Byte	В данном поле содержится четвертый байт MAC Address. Соответствует четвертому передаваемому байту поля DA
7:0	MAC_Address4_Third_Byte	В данном поле содержится третий байт MAC Address. Соответствует первому передаваемому байту поля DA

27.3.36 Регистр MAC_Address4_Low_Register

Регистр младшей части MAC адреса#4 содержит младшие два байта MAC адреса#4, как представлено в каноническом формате. Канонический формат показывает, в какой последовательности байты передаются на интерфейс Ethernet. Регистры старшей части MAC адреса#4 и средней части MAC адреса#4 содержат остаток 48-битного MAC адреса#4.

Формат регистра MAC_Address4_Low_Register приведен в Таблица 27.37.

Таблица 27.37. Формат регистра MAC_Address4_Low_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:8	MAC_Address4_Sixth_Byte	В данном поле содержится шестой байт MAC Address. Соответствует шестому передаваемому байту поля DA
7:0	MAC_Address4_Fifth_Byte	В данном поле содержится пятый байт MAC Address. Соответствует пятому передаваемому байту поля DA

27.3.37 Регистр MAC_Hash_Table1_Register

Регистр Multicast хеш-таблицы#1 формирует 64-битную Multicast Хеш-таблицу, которая используется для фильтрации Multicast адресов. Во время хеш-фильтрации содержимое поля Destination Address во входящем фрейме проходят через CRC логику, а верхние 6 бит CRC регистра используются в качестве индекса Хеш-таблицы. Значение *6'b000000* выбирает 0-й бит Хеш-таблицы, значение *6'b111111* выбирает 63-й бит Хеш-таблицы. Если Multicast фрейм - хеширован в Хеш-таблице и соответствующий бит в Хеш-таблице равен «1», то Multicast фрейм принимается, иначе - отклоняется.

Формат регистра MAC_Hash_Table1_Register приведен в Таблица 27.38.

Таблица 27.38. Формат регистра MAC_Hash_Table1_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:0	Hash_Table_Bits_15_0	Это поле определяет младшие 16 бит Multi-Cast Hash таблицы.

27.3.38 Регистр MAC_Hash_Table2_Register

Регистр Multicast хеш-таблицы#2 формирует 64-битную Multicast Хеш-таблицу, которая используется для фильтрации Multicast адресов. Во время хеш-фильтрации содержимое поля Destination Address во входящем фрейме проходят через CRC логику, а верхние 6 бит CRC регистра используются в качестве индекса Хеш-таблицы. Значение *6'b000000* выбирает 0-й бит Хеш-таблицы, значение *6'b111111* выбирает 63-й бит Хеш-таблицы. Если Multicast фрейм - хеширован в Хеш-таблице и соответствующий бит в Хеш-таблице равен «1», то Multicast фрейм принимается, иначе - отклоняется.

Формат регистра MAC_Hash_Table2_Register приведен в Таблица 27.39.

Таблица 27.39. Формат регистра MAC_Hash_Table2_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:0	Hash_Table_Bits_31_16	Это поле определяет биты [31:16] Multi-Cast Hash таблицы.

27.3.39 Регистр MAC_Hash_Table3_Register

Регистр Multicast хеш-таблицы#3 формирует 64-битную Multicast Хеш-таблицу, которая используется для фильтрации Multicast адресов. Во время хеш-фильтрации содержимое поля Destination Address во входящем фрейме проходят через CRC логику, а верхние 6 бит CRC регистра используются в качестве индекса Хеш-таблицы. Значение *6'b000000* выбирает 0-й бит Хеш-таблицы, значение *6'b111111* выбирает 63-й бит Хеш-таблицы. Если Multicast фрейм - хеширован в Хеш-таблице и соответствующий бит в Хеш-таблице равен «1», то Multicast фрейм принимается, иначе - отклоняется.

Формат регистра MAC_Hash_Table3_Register приведен в Таблица 27.40.

Таблица 27.40. Формат регистра MAC_Hash_Table3_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:0	Hash_Table_Bits_47_32	Это поле определяет младшие биты [47:32] Multi-Cast Hash таблицы.

27.3.40 Регистр MAC_Hash_Table4_Register

Регистр Multicast хеш-таблицы#4 формирует 64-битную Multicast Хеш-таблицу, которая используется для фильтрации Multicast адресов. Во время хеш-фильтрации содержимое поля Destination Address во входящем фрейме проходят через CRC логику, а верхние 6 бит CRC регистра используются в качестве индекса Хеш-таблицы. Значение *6'b000000* выбирает 0-й бит Хеш-таблицы, значение *6'b111111* выбирает 63-й бит Хеш-таблицы. Если Multicast фрейм - хеширован в Хеш-таблице и соответствующий бит в Хеш-таблице равен «1», то Multicast фрейм принимается, иначе - отклоняется.

Формат регистра MAC_Hash_Table4_Register приведен в Таблица 27.41.

Таблица 27.41. Формат регистра MAC_Hash_Table4_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:0	Hash_Table_Bits_63_48	Это поле определяет младшие биты [63:48] Multi-Cast Hash таблицы.

27.3.41 Регистр MAC_FlowControl_Control_Register

Регистр управления потоком данных MAC управляет различными особенностями потока данных (генерирование/прием пакета PAUSE) GEMAC ядра, если контроллер работает в дуплексном режиме. Если GEMAC ядро работает в полудуплексном режиме, функциональные возможности управления потоком данных заблокированы, и принимаемые фреймы PAUSE обрабатываются как нормальные пакеты и не декодируются.

Формат регистра MAC_FlowControl_Control_Register приведен в Таблица 27.42.

Таблица 27.42. Формат регистра MAC_FlowControl_Control_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:8	Enabled_Priorities	В этом поле указываются приоритеты, которые включены для генерации и обнаружения управления потоком Priority PAUSE. Управляющие сигналы XOFF / XON от пользовательской логики контролируются только для включенных Приоритетов. На приеме отслеживаются только включенные Приоритеты для поля Time-vector.
7	-	Резерв
6	Priority_PAUSE_Decode_Enable	Если бит установлен, то контроллер декодирует все входфщие пакеты Priority PAUSE Control в соответствии со спецификацией IEEE 802.3. Если контроллер принял достоверный пакет Priority PAUSE Control, он извлекает таймеры для включенных приоритетов, и соответствующее уведомление XOFF генерируется для логики пользователя, пока счетчик не равен нулю. Если бит сброшен, то контроллер не декодирует пакеты Priority PAUSE Control Этот бит действителен только тогда, когда ядро GEMAC работает в дуплексном режиме.
5	Priority_FlowControl_Generation_Enable	Когда установлено, GEMAC Core включен для передачи Priority PAUSE Control кадров. Кадры Priority PAUSE Control передаются либо с помощью программного управления, либо с внешних элементов управления Priority XON/XOFF. Параметры для сгенерированного Priority PAUSE кадра управляются отдельно. При сбросе ядро GEMAC не создает Priority PAUSE Control кадр. Этот бит действителен только тогда, когда ядро GEMAC работает в дуплексном режиме. Примечание. Функции PAUSE и Priority PAUSE являются взаимоисключающими. Таким образом, программное обеспечение должно включать только один.
4	Block_Pause_Frames	Если бит установлен, то принятый управляющий пакет PAUSE декодируется, но не передается в память. Эти кадры удаляются из Receive FIFO. Если бит сброшен, то принятый управляющий пакет PAUSE декодируется и передается в память как любой другой пакет.
3	FlowControl_MultiCast_Mode	Если данный бит установлен, то в поле DA, сгенерированного пакета управления потоком PAUSE, устанавливается зарезервированный Multi-Cast адрес (01:80:C2:00:00:01). Если бит сброшен, то в поле DA, сгенерированного пакета управления потоком PAUSE, устанавливается значение, запрограммированное в регистрах MAC Flow-Control Destination Address High/Med/Low Registers.

Номер бита	Условное обозначение	Назначение
2	Auto_Flow_Control_Generation_Enable	Если бит установлен, то контроллер генерирует автоматически управляющий пакет PAUSE при пересечении верхнего или нижнего уровня заполнения Receive FIFO. Если произошло пересечение верхнего уровня заполнения (снизу вверх), то в поле PauseTime пакета PAUSE помещается значение регистра MAC Auto High Pause Time Register. Если произошло пересечение нижнего уровня заполнения (сверху вниз), то в поле PauseTime пакета PAUSE помещается значение регистра MAC Auto Low Pause Time Register.
1	FlowControl_Generation_Enable	Если бит установлен, то включается режим передачи управляющего пакета PAUSE. Пакет PAUSE может передаваться как по программному запросу, так и автоматически, если включен режим автоматической генерации при пересечении верхнего или нижнего уровня заполнения Receive FIFO. Параметры генерируемого пакета PAUSE контролируются отдельно. Если бит сброшен, то ядро не генерирует управляющий пакет PAUSE. Использование этого бита возможно только при работе в Full-Duplex режиме.
0	FlowControl_Decode_Enable	Если бит установлен, то в контроллере включаются операции управления потоком, и декодируются все входящие пакеты PAUSE в соответствии со спецификацией IEEE 802.3. Если ядро принимает контролирующий пакет PAUSE, то передача кадров приостанавливается на время, указанное в поле PAUSE_TIME принятого пакета PAUSE. Если бит сброшен, то операции управления потоком выключены, контроллер не декодирует пакеты PAUSE. Использование этого бита возможно только при работе в Full-Duplex режиме.

27.3.42 Регистр MAC_FlowControl_PriPause_Frame_Generate_Register

Регистр управления генерированием пакета PAUSE используется для программного управления процессом генерирования фрейма PAUSE. Содержание сгенерированного пакета PAUSE берется из различных регистров управления потоком данных. Данный регистр обеспечивает механизм установления связи между программным обеспечением и GEMAC ядром, генерирующим фрейм PAUSE, и отображает признак завершения формирования фрейма PAUSE.

Формат регистра MAC_FlowControl_PriPause_Frame_Generate_Register приведен в Таблица 27.43.

Таблица 27.43. Формат регистра MAC_FlowControl_PriPause_Frame_Generate_Register

Номер бита	Условное обозначение	Назначение
31:2	-	Резерв
1	Generate_Priority_PAUSE_Frame	Установка данного бита информирует ядро о необходимости сформировать управляющий пакет Priority PAUSE. Перед установкой этого бита необходимо убедиться, что он сброшен. Если бит установлен, то ядро старается как можно быстрее сформировать пакет Priority PAUSE Control Если передатчик MAC находится в состоянии IDLE, то пакет генерируется немедленно,

Номер бита	Условное обозначение	Назначение
		если передатчик MAC в данный момент передает кадр пользовательских данных, генерация Priority PAUSE Control переводится в режим ожидания до тех пор, пока передача текущего кадра не будет завершена, и не будет удовлетворен IFG для следующего кадра. в этот момент передается Priority PAUSE Control. Программное обеспечение должно непрерывно опрашивать этот бит, чтобы увидеть, завершена или нет передача Priority PAUSE кадра. Когда передается Priority PAUSE control кадр, GEMAC очищает этот бит. Сгенерированный кадр Priority PAUSE будет основан на разрешенных приоритетах. Этот бит действителен только в том случае, если ядро GEMAC работает в Full-Duplex режиме и установлен бит «Priority PAUSE Flow-Control Generation Enable» в регистре «MAC Flow-Control».
0	Generate_PAUSE_ Frame	Установка данного бита информирует ядро о необходимости сформировать управляющий пакет PAUSE. Перед установкой этого бита необходимо убедиться, что он сброшен. Если бит установлен, то ядро старается как можно быстрее сформировать пакет PAUSE. Если передатчик MAC находится в состоянии IDLE, то пакет PAUSE транслируется немедленно. Если передатчик MAC в текущий момент осуществляет передачу, то трансляция пакета PAUSE откладывается до завершения передачи и выдержки IFG интервала. Как только передача управляющего пакета PAUSE завершена, бит сбрасывается. Использование этого бита возможно только при работе в Full-Duplex режиме и установке бита Flow-Control Generation Enable в регистре MAC Flow-Control Register.

27.3.43 Регистр MAC_FlowControl_Source_Address_High_Register

Регистр управления верхней частью адреса источника содержит первые два байта Source Address (как представлено в Каноническом Формате), используемых в генерируемых фреймах PAUSE. Канонический формат показывает, в какой последовательности байты передаются на интерфейс Ethernet.

Формат регистра MAC_FlowControl_Source_Address_High_Register приведен в Таблица 27.44.

Таблица 27.44. Формат регистра MAC_FlowControl_Source_Address_High_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:8	MAC_FlowControl _Source_Address_ Second_Byte	В данном поле содержится второй байт Source Address, использующегося при формировании управляющего пакета PAUSE. Это второй байт поля SA пакета PAUSE Flow-Control.
7:0	MAC_FlowControl _Source_Address_ First_Byte	В данном поле содержится первый байт Source Address, использующегося при формировании управляющего пакета PAUSE. Это первый байт поля SA пакета PAUSE Flow-Control.

27.3.44 Регистр MAC_FlowControl_Source_Address_Med_Register

Регистр управления средней частью адреса источника содержит средние два байта Source Address (как представлено в Каноническом Формате), используемых в генерируемых фреймах PAUSE. Канонический формат показывает, в какой последовательности байты передаются на интерфейс Ethernet.

Формат регистра MAC_FlowControl_Source_Address_Med_Register приведен в Таблица 27.45.

Таблица 27.45. Формат регистра MAC_FlowControl_Source_Address_Med_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:8	MAC_FlowControl_Source_Address_Fourth_Byte	В данном поле содержится четвертый байт Source Address, использующегося при формировании управляющего пакета PAUSE. Это четвертый байт поля SA пакета PAUSE Flow-Control.
7:0	MAC_FlowControl_Source_Address_Third_Byte	В данном поле содержится третий байт Source Address, использующегося при формировании управляющего пакета PAUSE. Это третий байт поля SA пакета PAUSE Flow-Control.

27.3.45 Регистр MAC_FlowControl_Source_Address_Low_Register

Регистр управления нижней частью адреса источника содержит последние два байта Source Address (как представлено в Каноническом Формате), используемых в генерируемых фреймах PAUSE. Канонический формат показывает, в какой последовательности байты передаются на интерфейс Ethernet.

Формат регистра MAC_FlowControl_Source_Address_Low_Register приведен в Таблица 27.46.

Таблица 27.46. Формат регистра MAC_FlowControl_Source_Address_Low_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:8	MAC_FlowControl_Source_Address_Sixth_Byte	В данном поле содержится шестой байт Source Address, использующегося при формировании управляющего пакета PAUSE. Это шестой байт поля SA пакета PAUSE Flow-Control.
7:0	MAC_FlowControl_Source_Address_Fifth_Byte	В данном поле содержится пятый байт Source Address, использующегося при формировании управляющего пакета PAUSE. Это пятый байт поля SA пакета PAUSE Flow-Control.

27.3.46 Регистр MAC_FlowControl_Destination_Address_High_Register

Регистр управления верхней частью адреса назначения содержит первые два байта Destination Address (как представлено в Каноническом Формате), используемых в генерируемых фреймах PAUSE (только при работе с Unicast адресами). Канонический формат показывает, в какой последовательности байты передаются на интерфейс Ethernet.

Формат регистра MAC_FlowControl_Destination_Address_High_Register приведен в Таблица 27.47.

Таблица 27.47. Формат регистра MAC_FlowControl_Destination_Address_High_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:8	MAC_FlowControl_Destination_Address_Second_Byte	В данном поле содержится второй байт Destination Address, использующегося при формировании управляющего пакета (Priority) PAUSE. Это второй байт поля DA пакета (Priority) PAUSE Flow-Control.
7:0	MAC_FlowControl_Destination_Address_First_Byte	В данном поле содержится первый байт Destination Address, использующегося при формировании управляющего пакета (Priority) PAUSE. Это первый байт поля DA пакета (Priority) PAUSE Flow-Control.

27.3.47 Регистр MAC_FlowControl_Destination_Address_Med_Register

Регистр управления средней частью адреса назначения содержит средние два байта Destination Address (как представлено в Каноническом Формате), используемых в генерируемых фреймах PAUSE (только при работе с Unicast адресами). Канонический формат показывает, в какой последовательности байты передаются на интерфейс Ethernet.

Формат регистра MAC_FlowControl_Destination_Address_Med_Register приведен в Таблица 27.48.

Таблица 27.48. Формат регистра MAC_FlowControl_Destination_Address_Med_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:8	MAC_FlowControl_Destination_Address_Fourth_Byte	В данном поле содержится четвертый байт Destination Address, использующегося при формировании управляющего пакета (Priority) PAUSE. Это четвертый байт поля DA пакета (Priority) PAUSE Flow-Control.
7:0	MAC_FlowControl_Destination_Address_Third_Byte	В данном поле содержится третий байт Destination Address, использующегося при формировании управляющего пакета (Priority) PAUSE. Это третий байт поля DA пакета (Priority) PAUSE Flow-Control.

27.3.48 Регистр MAC_FlowControl_Destination_Address_Low_Register

Регистр управления нижней частью адреса назначения содержит младшие два байта Destination Address (как представлено в Каноническом Формате), используемых в генерируемых фреймах PAUSE (только при работе с Unicast адресами). Канонический формат показывает, в какой последовательности байты передаются на интерфейс Ethernet.

Формат регистра MAC_FlowControl_Destination_Address_Low_Register приведен в Таблица 27.49.

Таблица 27.49. Формат регистра MAC_FlowControl_Destination_Address_Low_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:8	MAC_FlowControl_Destination_Address_Sixth_Byte	В данном поле содержится шестой байт Destination Address, использующегося при формировании управляющего пакета (Priority) PAUSE. Это шестой байт поля DA пакета (Priority) PAUSE Flow-Control.
7:0	MAC_FlowControl_Destination_Address_Fifth_Byte	В данном поле содержится пятый байт Destination Address, использующегося при формировании управляющего пакета (Priority) PAUSE. Это пятый байт поля DA пакета (Priority) PAUSE Flow-Control.

27.3.49 Регистр MAC_FlowControl_Pause_Time_Value_Register

MAC Flow-Control Pause Time Value Register содержит значение Pause-Time, использующееся в сгенерированном пакете PAUSE, если установлен бит Generate (Priority) Pause Frame в регистре MAC Flow-Control (Priority) PAUSE Frame Generate Register.

Формат регистра MAC_FlowControl_Pause_Time_Value_Register приведен в Таблица 27.50.

Таблица 27.50. Формат регистра MAC_FlowControl_Pause_Time_Value_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:0	MAC_FlowControl_Pause_Time	В данном поле содержится значение Pause-Time, использующееся в сгенерированном пакете (Priority) PAUSE.

27.3.50 Регистр MAC_FlowControl_Auto_Gen_Hi_Pause_Time_Value_Register

Регистр управления значением времени автоматической паузы содержит значение времени паузы, которое используется при генерировании фрейма PAUSE, если данные в Receive FIFO превысили уровень, установленный регистром Auto Gen Hi Threshold Register.

Формат регистра MAC_FlowControl_Auto_Gen_Hi_Pause_Time_Value_Register приведен в Таблица 27.51.

Таблица 27.51. Формат регистра MAC_FlowControl_Auto_Gen_Hi_Pause_Time_Value_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:0	MAC_FlowControl_Auto_Gen_Hi_Pause_Time	В данном поле содержится значение Pause-Time, использующееся в сгенерированном пакете PAUSE, если пакет сгенерирован при переполнении Receive FIFO.

27.3.51 Регистр MAC_FlowControl_Auto_Gen_Lo_Pause_Time_Value_Register

Регистр управления значения времени автоматической паузы содержит значение времени паузы, которое используется при генерировании фрейма PAUSE, когда объем данных в Receive FIFO опустился ниже значения, прописанного в регистре Auto Gen Low Threshold Register, при условии, что до этого было превышение уровня, установленного регистром Auto Gen Hi Threshold Register.

Формат регистра MAC_FlowControl_Auto_Gen_Lo_Pause_Time_Value_Register приведен в Таблица 27.52.

Таблица 27.52. Формат регистра MAC_FlowControl_Auto_Gen_Lo_Pause_Time_Value_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:0	MAC_FlowControl_Pause_Time	В данном поле содержится значение Pause-Time, использующееся в сгенерированном пакете PAUSE, если пакет сгенерирован при опустошении Receive FIFO (пересечении границы Low Threshold после переполнения с пересечением границы High Threshold).

27.3.52 Регистр MAC_Flow_Control_Auto_Pause_Frame_Gen_Hi_Threshold_Register

MAC Flow-Control Auto Pause Frame Gen Hi Threshold Register содержит пороговое значение Receive FIFO, превышение которого влечет формирование пакета PAUSE со значением времени, запрограммированном в регистре MAC Flow-Control Auto Gen Hi Pause Time Value Register.

Формат регистра MAC_Flow_Control_Auto_Pause_Frame_Gen_Hi_Threshold_Register приведен в Таблица 27.53.

Таблица 27.53. Формат регистра MAC_Flow_Control_Auto_Pause_Frame_Gen_Hi_Threshold_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:0	MAC_FlowControl_Auto_Pause_Frame_Gen_Hi_Threshold	В данном поле содержится пороговое значение Receive FIFO, превышение которого влечет формирование пакета PAUSE со значением времени, запрограммированном в регистре MAC Flow-Control Auto Gen Hi Pause Time Value Register.

27.3.53 Регистр MAC_Flow_Control_Auto_Pause_Frame_Gen_Low_Threshold_Register

MAC Flow-Control Auto Pause Frame Gen Lo Threshold Register содержит пороговое значение Receive FIFO. При понижении количества данных в FIFO ниже этого порога

формируется пакет PAUSE со значением времени, запрограммированном в регистре MAC Flow-Control Auto Gen Lo Pause Time Value Register.

Формат регистра MAC_Flow_Control_Auto_Pause_Frame_Gen_Low_Threshold_Register приведен в Таблица 27.54.

Таблица 27.54. Формат регистра MAC_Flow_Control_Auto_Pause_Frame_Gen_Low_Threshold_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:0	MAC_FlowControl_Auto_Pause_Frame_Gen_Lo_Threshold	В данном поле содержится пороговое значение Receive FIFO. При понижении количества данных в FIFO ниже этого порога формируется пакет PAUSE со значением времени запрограммированном в регистре MAC Flow-Control Auto Gen Lo Pause Time Value Register.

27.3.54 Регистр MAC_MDIO_Control_Register

Регистр управления MDIO используется для управления формированием MDIO пакетов к внешнему чипу PHY. Этот регистр содержит различные поля MDIO пакета, такие как адрес PHY, адрес регистра и т.д.

Формат регистра MAC_MDIO_Control_Register приведен в Таблица 27.55.

Таблица 27.55. Формат регистра MAC_MDIO_Control_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15	Start_MDIO_Transaction	Установка этого бита ведет к формированию MDIO пакета. Перед установкой бита необходимо убедиться, что он очищен. Когда бит установлен – контроллер формирует пакет, используя ранее запрограммированные поля данного регистра. Если осуществляется запись, то регистр MDIO Data Register используется в качестве источника данных. Если осуществляется чтение, то регистр MDIO Data Register используется в качестве приемника данных. Если бит сбросился, то это означает, что транзакция завершена. При транзакции записи необходимо прописать записываемые данные в регистр MDIO Data Register перед установкой бита старта. При транзакции чтения необходимо прочитать данные из регистра MDIO Data Register после снятия бита старта.
14:11	-	Резерв
10	MDIO_Read_Write	Данный бит определяет тип обмена, который будет осуществлен при старте MDIO транзакции. 1’b1: чтение 1’b0: запись
9:5	Register_Address	Данные биты используются для формирования поля Register Address пакета MDIO.
4:0	PHY_Address	Данные биты используются для формирования поля PHY Address пакета MDIO.

27.3.55 Регистр MAC_MDIO_Data_Register

Регистр данных MDIO содержит 16-разрядные данные, которые будут записаны в регистр PHY, в случае подачи команды записи MDIO либо 16-разрядные прочитанные данные из регистра PHY, в случае подачи команды чтения MDIO.

Формат регистра MAC_MDIO_Data_Register приведен в Таблица 27.56.

Таблица 27.56. Формат регистра MAC_MDIO_Data_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:0	MDIO_Data	Данное поле содержит 16-битовое значение, прочитанное из PHY после операции чтения, или 16-битовое значение, записываемое PHY при операции записи.

27.3.56 Регистр MAC_Rx_StatCtr_Control_Register

Регистр статистики приемника используется для отображения статистических счетчиков приемника GEMAC ядра. Номер читаемого регистра необходимо запрограммировать в соответствующем поле данного регистра. Как только операция чтения будет закончена, прочитанные данные отобразятся в регистрах MAC Receive StatCtr. Data High/Low Registers.

Формат регистра MAC_Rx_StatCtr_Control_Register приведен в Таблица 27.57.

Таблица 27.57. Формат регистра MAC_Rx_StatCtr_Control_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15	Start_Receive_Counter_Read	Установка данного бита является командой контроллеру о необходимость прочитать регистр статистики из внутренней памяти. Перед установкой данного бита необходимо убедиться, что он сброшен. По состоянию данного бита можно определить статус выполняемой операции чтения. Сброс бита означает, что операция чтения выполнена. Для того чтобы узнать значение читаемого регистра статистики, необходимо, после сброса бита, прочитать регистры MAC Receive StatCtr. Data High/Low Registers.
14:5	-	Резерв
4:0	Receive_Counter_Number	Данное поле содержит номер читаемого счетчика статистики приема.

27.3.57 Регистр MAC_Rx_StatCtr_Data_High_Register

Регистр старшей части статистических данных приемника содержит верхние 16 бит 32-разрядных данных статистики

Формат регистра MAC_Rx_StatCtr_Data_High_Register приведен в Таблица 27.58.

Таблица 27.58. Формат регистра MAC_Rx_StatCtr_Data_High_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:0	Receive_StatCtr_Data_High	Данное поле содержит старшие 16 бит 32-разрядного Receive Counter, прочитанного в предыдущей операции.

27.3.58 Регистр MAC_Rx_StatCtr_Data_Low_Register

Регистр младшей части статистических данных приемника содержит нижние 16 бит 32-разрядных данных статистики.

Формат регистра MAC_Rx_StatCtr_Data_Low_Register приведен в Таблица 27.59.

Таблица 27.59. Формат регистра MAC_Rx_StatCtr_Data_Low_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:0	Receive_StatCtr_Data_Low	Данное поле содержит младшие 16 бит 32-разрядного Receive Counter, прочитанного в предыдущей операции.

27.3.59 Регистр MAC_Tx_StatCtr_Control_Register

Регистр статистики передатчика используется для отображения статистических счетчиков передатчика GEMAC ядра. Номер читаемого регистра необходимо запрограммировать в соответствующем поле данного регистра. Как только операция чтения будет закончена, прочитанные данные отобразятся в регистрах MAC Transmit StatCtr. Data High/Low Registers.

Формат регистра MAC_Tx_StatCtr_Control_Register приведен в Таблица 27.60.

Таблица 27.60. Формат регистра MAC_Tx_StatCtr_Control_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15	Start_Receive_Counter_Read	Установка данного бита является командой контроллеру о необходимость прочитать регистр статистики из внутренней памяти. Перед установкой данного бита необходимо убедиться, что он сброшен. По состоянию данного бита можно определить статус выполняемой операции чтения. Сброс бита означает, что операция чтения выполнена. Для того чтобы узнать значение читаемого регистра статистики, необходимо, после сброса бита, прочитать регистры MAC Transmit StatCtr. Data High/Low Registers.
14:5	-	Резерв
4:0	Transmit_Counter_Number	Данное поле содержит номер читаемого счетчика статистики передачи.

27.3.60 Регистр MAC_Tx_StatCtr_Data_High_Register

Регистр старшей части статистических данных передатчика содержит верхние 16 бит 32-разрядных данных статистики.

Формат регистра MAC_Tx_StatCtr_Data_High_Register приведен в Таблица 27.61.

Таблица 27.61. Формат регистра MAC_Tx_StatCtr_Data_High_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:0	Transmit_StatCtr_Data_High	Данное поле содержит старшие 16 бит 32-разрядного Transmit Counter, прочитанного в предыдущей операции.

27.3.61 Регистр MAC_Tx_StatCtr_Data_Low_Register

Регистр младшей части статистических данных передатчика содержит нижние 16 бит 32-разрядных данных статистики.

Формат регистра MAC_Tx_StatCtr_Data_Low_Register приведен в Таблица 27.62.

Таблица 27.62. Формат регистра MAC_Tx_StatCtr_Data_Low_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:0	Transmit_StatCtr_Data_Low	Данное поле содержит младшие 16 бит 32-разрядного Transmit Counter, прочитанного в предыдущей операции.

27.3.62 Регистр MAC_Transmit_FIFO_Almost_Full_Register

Регистр порогового значения полноты FIFO передатчика содержит пороговое значение, которое используется Transmit FIFO для генерирования «AlmostFull» события к DMA. Это укажет DMA о необходимости приостановить передачу данных от памяти FIFO. Значение, запрограммированное в регистре - количество пустых ячеек, при котором будет формироваться событие «AlmostFull».

Формат регистра MAC_Transmit_FIFO_Almost_Full_Register приведен в Таблица 27.63.

Таблица 27.63. Формат регистра MAC_Transmit_FIFO_Almost_Full_Register

Номер бита	Условное обозначение	Назначение
31:14	-	Резерв
13:0	Transmit_FIFO_AlmostFull_Threshold	Данное поле содержит количество свободных позиций (32-разрядных слов) оставшихся в Transmit FIFO при котором необходимо сформировать событие AlmostFull. Как только данное событие сформировалось, DMA перестает подгружать следующие данные из памяти. Для правильной работы контроллера значение, запрограммированное в данном поле, должно вычисляться по формуле:

Номер бита	Условное обозначение	Назначение
		Transmit FIFO AlmostFull Threshold = FIFO_SIZE-14'h0008. Здесь FIFO_SIZE – глубина Transmit FIFO в 32-разрядных словах (FIFO_SIZE = 8192)

27.3.63 Регистр MAC_Transmit_Packet_Start_Threshold_Register

Регистр контроля начала передачи кадра в канал определяет пороговое значение в байтах заполненности Transmit FIFO. Когда количество данных в Transmit FIFO превышает заданную границу – стартует передача пакета на интерфейс RGMII. GEMAC ядро ждет заполнения FIFO до порогового значения или конца пакета в Transmit FIFO, прежде чем инициализировать передачу кадра на интерфейс RGMII. Грамотно выбранное значение данного регистра предотвращает переполнение Transmit FIFO и увеличивает пропускную способность шины.

Формат регистра MAC_Transmit_Packet_Start_Threshold_Register приведен в Таблица 27.64.

Таблица 27.64. Формат регистра MAC_Transmit_Packet_Start_Threshold_Register

Номер бита	Условное обозначение	Назначение
31:14	-	Резерв
13:0	Transmit_Packet_Start_Threshold	Данное поле отображает, какое количество байт должно быть помещено в Transmit FIFO для начала передачи в RGMII канал. Для начала передачи ядро ждет заполнения FIFO до указанной границы или признака конца пакета (End of Packet). Высокое значение данного поля уменьшает вероятность возникновения ситуаций, при которых DMA не успевает подгружать передаваемые данные из памяти. Также выбор границы заполнения должен основываться на скорости Ethernet интерфейса. Рекомендуются следующие значения для различных скоростей передачи: 10 Mbps: 64 байта 100 Mbps: 128 байта 1000 Mbps: 1024 байта Если записано значение 1518, то передатчик работает в режиме Store and Forward для всех пакетов независимо от скорости интерфейса.

27.3.64 Регистр MAC_Receive_Packet_Start_Threshold_Register

Регистр контроля начала передачи кадра в память определяет пороговое значение в байтах заполненности Receive FIFO. Когда количество данных в Receive FIFO превышает заданную границу - стартует передача принятого пакета в память. Если полученный пакет меньше чем значение, запрограммированное в этом регистре, пакет сбрасывается из FIFO и не передается памяти. Это исключает формирование маленьких транзакций, что также ведет к увеличению пропускной способности шины.

Формат регистра MAC_Receive_Packet_Start_Threshold_Register приведен в Таблица 27.65.

Таблица 27.65. Формат регистра MAC_Receive_Packet_Start_Threshold_Register

Номер бита	Условное обозначение	Назначение
31:14	-	Резерв
13:0	Receive_Packet_Start_Threshold	Данное поле отображает, какое количество байт должно быть принято в Receive FIFO для начала передачи данных в память. Пакеты, размер которых меньше данного значения, понимаются как Rant/Fragment, сбрасываются из FIFO и не передаются в память. Если установлено значение 64, то от передачи в память отфильтровываются все Runt пакеты. Значение 0 в данном поле конфигурирует контроллер на передачу в память всех пакетов независимо от размера и поля DA. Для включения адресной фильтрации необходимо занести значение не менее 12. Если установлен бит Store and Forward регистра Mac Receive Control Register, то значение, записанное в данном регистре, игнорируется.

27.3.65 Регистр MAC_Transmit_FIFO_Almost_Empty_Threshold_Register

Регистр порогового значения опустошения Transmit FIFO содержит значение, которое используется Transmit FIFO для формирования «AlmostEmpty» события к DMA. Значение, запрограммированное в регистре - количество занятых ячеек при котором будет формироваться событие «AlmostEmpty».

Формат регистра MAC_Transmit_FIFO_Almost_Empty_Threshold_Register приведен в Таблица 27.66.

Таблица 27.66. Формат регистра MAC_Transmit_FIFO_Almost_Empty_Threshold_Register

Номер бита	Условное обозначение	Назначение
31:8	-	Резерв
7:0	Transmit_FIFO_AlmostEmpty_Threshold	Данное поле содержит количество занятых позиций (32-разрядных или 64-разрядных слов) в Transmit FIFO при котором необходимо сформировать событие AlmostEmpty.

27.3.66 Регистр MAC_Transmit_FIFO_Space_Available_Hi_Threshold_Register

В настоящее время это значение не используется DMA и не требует программирования.

Формат регистра MAC_Transmit_FIFO_Space_Available_Hi_Threshold_Register приведен в Таблица 27.67.

Таблица 27.67. Формат регистра MAC_Transmit_FIFO_Space_Available_Hi_Threshold_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:0	Transmit_FIFO_Space_Available_Hi_Threshold	В настоящее время это значение не используется DMA и не требует программирования.

27.3.67 Регистр MAC_Transmit_FIFO_Space_Available_Lo_Threshold_Register

В настоящее время это значение не используется DMA и не требует программирования.

Формат регистра MAC_Transmit_FIFO_Space_Available_Lo_Threshold_Register приведен в Таблица 27.68.

Таблица 27.68. Формат регистра MAC_Transmit_FIFO_Space_Available_Lo_Threshold_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:0	Transmit_FIFO_Space_Available_Lo_Threshold	В настоящее время это значение не используется DMA и не требует программирования.

27.3.68 Регистр MAC_Receive_FIFO_Packet_Available_Threshold1_Register

В настоящее время это значение не используется DMA и не требует программирования.

Формат регистра MAC_Receive_FIFO_Packet_Available_Threshold1_Register приведен в Таблица 27.69.

Таблица 27.69. Формат регистра MAC_Receive_FIFO_Packet_Available_Threshold1_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:8	Receive_FIFO_Packet_Available_Threshold2	В настоящее время это значение не используется DMA и не требует программирования.

27.3.69 Регистр MAC_Receive_FIFO_Packet_Available_Threshold2_Register

В настоящее время это значение не используется DMA и не требует программирования.

Формат регистра MAC_Receive_FIFO_Packet_Available_Threshold2_Register приведен в Таблица 27.70.

Таблица 27.70. Формат регистра MAC_Receive_FIFO_Packet_Available_Threshold2_Register

Номер бита	Условное обозначение	Назначение
31:5	-	Резерв
4	Receive_FIFO_Packet_Available_Threshold_Selection	В настоящее время это значение не используется DMA и не требует программирования.
3:0	Receive_FIFO_Packet_Count_Threshold	В настоящее время это значение не используется DMA и не требует программирования.

27.3.70 Регистр MAC_Status_and_IRQ_Register

Регистр статуса и запроса на прерывание отображает информацию о состоянии GEMAC ядра и статусы запросов на прерывание. Биты запроса на прерывание используются для формирования запроса на прерывание CPU.

Формат регистра MAC_Status_and_IRQ_Register приведен в Таблица 27.71.

Таблица 27.71. Формат регистра MAC_Status_and_IRQ_Register

Номер бита	Условное обозначение	Назначение
31:2	-	Резерв
1	MAC_Jabber_IRQ	Установка данного бита говорит о том, что обнаружена передача в канале RGMII бессмысленных данных. Длина пакета больше, чем запрограммировано в регистре Transmit Jabber Count Register. MAC обрывает пакет, выставляет на RGMII интерфейс ошибку EOP/ERR, а поступающие из памяти данные игнорирует. Запрос на прерывание сбрасывается записью единицы.
0	MAC_Underrun_IRQ	Установка данного бита говорит о том, что передача пакета в RGMII канал была прекращена из-за того, что DMA не успевает подгружать данные из памяти. MAC выставляет на RGMII интерфейс ошибку EOP/ERR, а поступающие из памяти данные игнорирует. Запрос на прерывание сбрасывается записью единицы.

27.3.71 Регистр MAC_Interrupt_Enable_Register

Регистр масок запросов на прерывание используется для управления различными запросами на прерывание.

Формат регистра MAC_Interrupt_Enable_Register приведен в Таблица 27.72.

Таблица 27.72. Формат регистра MAC_Interrupt_Enable_Register

Номер бита	Условное обозначение	Назначение
31:2	-	Резерв
1	Jabber_Interrupt_Enable	Разрешение запроса на прерывание MAC Jabber IRQ 1- прерывание разрешено 0- прерывание запрещено
0	MAC_Underrun_Interrupt_Enable	Разрешение запроса на прерывание MAC Underrun IRQ 1- прерывание разрешено

Номер бита	Условное обозначение	Назначение
		0- прерывание запрещено

27.3.72 Регистр MAC_VLAN_TPID1_Register

Регистр используются для настройки пользовательских значений VLAN TPID1, которые используются для обнаружения кадров VLAN при учете максимальной длины кадра. Значения по умолчанию достаточны для определения общих значений TPID VLAN, которые используются в типичном приложении.

Формат регистра MAC_VLAN_TPID1_Register приведен в Таблица 27.73.

Таблица 27.73. Формат регистра MAC_VLAN_TPID1_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:0	MAC_VLAN_TPID_1	Данное поле содержит первое из трех значений TPID, используемое для обнаружения VLAN поля.

27.3.73 Регистр MAC_VLAN_TPID2_Register

Регистр используются для настройки пользовательских значений VLAN TPID2, которые используются для обнаружения кадров VLAN при учете максимальной длины кадра. Значения по умолчанию достаточны для определения общих значений TPID VLAN, которые используются в типичном приложении.

Формат регистра MAC_VLAN_TPID2_Register приведен в Таблица 27.74.

Таблица 27.74. Формат регистра MAC_VLAN_TPID2_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:0	MAC_VLAN_TPID2	Данное поле содержит второе из трех значений TPID, используемое для обнаружения VLAN поля.

27.3.74 Регистр MAC_VLAN_TPID3_Register

Регистр используются для настройки пользовательских значений VLAN TPID3, которые используются для обнаружения кадров VLAN при учете максимальной длины кадра. Значения по умолчанию достаточны для определения общих значений TPID VLAN, которые используются в типичном приложении.

Формат регистра MAC_VLAN_TPID3_Register приведен в Таблица 27.75.

Таблица 27.75. Формат регистра MAC_VLAN_TPID3_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв

Номер бита	Условное обозначение	Назначение
15:0	MAC_VLAN_TPID3	Данное поле содержит третье из трех значений TPID, используемое для обнаружения VLAN поля.

27.3.75 Регистр Module_1588_Control_Register

Module_1588_Control_Register используется для управления основными функциями модуля 1588 и пакетов с метками времени.

Формат регистра Module_1588_Control_Register приведен в Таблица 27.76.

Таблица 27.76. Формат регистра Module_1588_Control_Register

Номер бита	Условное обозначение	Назначение
31:6	-	Резерв
5:3	RX_PTP_Packet_Type	Биты управления типом пакетов выбранных для таймстемпинга входного тракта. Если пакет принят без ошибок и совпали все запрограммированные критерии тогда метка времени принимается и передается ПО. Отслеживаются следующие типы PTP пакетов: 3'b000: Timestamp Version#2 PTP Packets in L2 Encapsulation (PTP Packets as L2 payload) only (Sync or Delay_req packet types based on the PTP MessageID) 3'b001: Timestamp Version#1 PTP Packets in L4 Encapsulation (PTP packets as UDP Payload) only (Sync or Delay_req packet types based on the PTP MessageID) 3'b010: Timestamp Version#2 PTP Packets in L2 or L4 Encapuslation (PTP packets as L2 payload or as UDP Payload) Packets in 3'b011 – 3'b111: Резерв
2	RX_Timestamping_Enable	Бит включения функции анализа меток времени входящих PTP пакетов, которые соответствуют запрограммированным критериям (MessageID, EtherType PTP, версия порта UDP и т.д.) 1 – включено 0 – выключено
1	TX_Timestamping_Enable	Бит включения функции проставления меток времени в исходящих PTP пакетах (под управление ПО через биты дескриптора) 1 – включено 0 – выключено
0	-	Резерв

27.3.76 Регистр Increment_Attributes_Register

Increment_Attributes_Register используется для управления приращением системного таймера.

Формат регистра Increment_Attributes_Register приведен в Таблица 27.77.

Таблица 27.77. Формат регистра Increment_Attributes_Register

Номер бита	Условное обозначение	Назначение
31:24	Increment_Period	Это поле определяет период для выполнения операции инкремента на системном таймере. Задается в тактах clk_1588 между операциями инкремента. Если значение равно 0x00, то операция увеличения будет отключена.

Номер бита	Условное обозначение	Назначение
23:0	Increment_Value	Это поле определяет значение, которое будет добавлено в системный таймер для каждой операции приращения. Это значение определяется гранулярностью и точностью внедренного системного таймера.

27.3.77 Регистр PTP_Ethertype_Register

PTP_Ethertype_Register используется для определения Ethertype пакетов, если используется L2 инкапсуляция.

Формат регистра PTP_Ethertype_Register приведен в Таблица 27.78.

Таблица 27.78. Формат регистра PTP_Ethertype_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:0	PTP_Ethertype	Это поле определяет поле Ethertype во входящих пакетах для мониторинга сообщений PTP, если пакеты PTP V2 используют инкапсуляцию L2. Значение этого регистра запрограммировано / считано в сетевом порядке. Типичный Ethertype, используемый для сообщений PTP, - 16'h88F7

27.3.78 Регистр PTP_MessageID_Register

PTP_Message_ID_Register используется для определения «PTP Message ID» входящих пакетов PTP, которые будут выбраны для таймстемпинга.

Формат регистра PTP_MessageID_Register приведен в Таблица 27.79.

Таблица 27.79. Формат регистра PTP_MessageID_Register

Номер бита	Условное обозначение	Назначение
31:8	-	Резерв
7:0	PTP_Message_ID	Это поле определяет PTP MessageID (или PTP Control для V1) во входящих PTP пакетах выбранных для таймстемпинга. Программное обеспечение может использовать это поле для отметки времени пакетов SYNC или DELAY_REQ в зависимости от того, является ли устройство ведомым или ведущим соответственно. При мониторинге сообщений PTP V1 это поле соответствует PTP Control, а при мониторинге сообщений PTP V2 это поле соответствует PTP MessageID.

27.3.79 Регистр PTP_UDP_Port_Register

PTP_UDP_Port_Register используется для определения номера порта назначения UDP пакетов, если сообщение PTP использует инкапсуляцию L4.

Формат регистра PTP_UDP_Port_Register приведен в Таблица 27.80.

Таблица 27.80. Формат регистра PTP_UDP_Port_Register

Номер бита	Условное обозначение	Назначение
31:16	-	Резерв
15:0	PTP_UDP_Port	Это поле задает поле назначения «UDP port» во входящих пакетах для мониторинга сообщений PTP, если пакеты PTP V1/V2 используют инкапсуляцию L4. Значение этого регистра запрограммировано/считано в сетевом порядке. Типичными значениями «UDP Ports», которые используются для сообщений PTP, являются 16'h013F или 16'h0140.

27.3.80 Регистр System_Time_Value_Lower_Register

System_Time_Value_ Lower_Register используются для отображения текущего значения системного таймера и содержит младшие 32 бита системного таймера.

Формат регистра System_Time_Value_Lower_Register приведен в Таблица 27.81.

Таблица 27.81. Формат регистра System_Time_Value_Lower_Register

Номер бита	Условное обозначение	Назначение
31:0	System_Time_Value_Lower	Младшие 32 бита значения системного таймера. ПО должно сначала прочитать этот регистр перед чтением регистра System_Time_Value_Upper_Register. Старшие 32 бита системного таймера фиксируются и загружаются в регистр System_Time_Value_Upper_Register, когда считывается этот регистр.

27.3.81 Регистр System_Time_Value_Upper_Register

System_Time_Value_Upper_Register используются для отображения текущего значения системного таймера и содержит старшие 32 бита системного таймера.

Формат регистра System_Time_Value_Upper_Register приведен в Таблица 27.82.

Таблица 27.82. Формат регистра System_Time_Value_Upper_Register

Номер бита	Условное обозначение	Назначение
31:0	System_Time_Value_Upper	Старшие 32 бита значения системного таймера. ПО должно читать этот регистр после чтения System_Time_Value_Lower_Register.

27.3.82 Регистр System_Time_Adjust_Control_Lower_Register

System_Time_Adjust_Control_Lower_Register регистр подстройки системного таймера. ПО должно запрограммировать System Timer Adjust Control Lower Register с параметрами подстройки, а затем запрограммировать верхний регистр для выполнения регулировки.

Формат регистра System_Time_Adjust_Control_Lower_Register приведен в Таблица 27.83.

Таблица 27.83. Формат регистра System_Time_Adjust_Control_Lower_Register

Номер бита	Условное обозначение	Назначение
31:0	System_Time_Adjust_Value	Это поле определяет 32-битное значение регулировки для системного таймера. Тип регулировки (положительный или отрицательный) и фактическая регулировка контролируются через System_Time_Adjust_Control_Upper_Register.

27.3.83 Регистр System_Time_Adjust_Control_Upper_Register

System_Time_Adjust_Control_Upper_Register используются для подстройки системного таймера. ПО должно запрограммировать System_Time_Adjust_Control_Lower_Register, а затем запрограммировать верхний регистр для выполнения регулировки.

Формат регистра System_Time_Adjust_Control_Upper_Register приведен в Таблица 27.84.

Таблица 27.84. Формат регистра System_Time_Adjust_Control_Upper_Register

Номер бита	Условное обозначение	Назначение
31:0	System_Time_Adjust_Control	WO регистр. Используется для выполнения подстройки системного времени. Регулировка происходит, в момент записи этого регистра. Тип настройки (положительный или отрицательный) определяется битом [31] этого регистра. Ниже приведен тип корректировки на основе значения бита [31]: 1'b0 – положительная корректировка. Значение, запрограммированное в System_Time_Adjust_Value добавляется к текущему системному таймеру. 1'b1 – отрицательная корректировки. Значение, запрограммированное в System_Time_Adjust_Value, вычитается из текущего системного таймера.

27.3.84 Регистр Transmit_Timestamp_Value_Lower_Register

Transmit_Timestamp_Registers используются для считывания текущей метки времени для указанного пакета передачи. Логика не будет обновлять это значение (не выполняет таймстемпинг для другого пакета), пока эти регистры не будут прочитаны. Регистры имеют RO доступ. Первый регистр содержит нижние 32 бита значения таймстемпа передачи, второй регистр содержит верхние 32 бита таймстемпа передачи.

Формат регистра Transmit_Timestamp_Value_Lower_Register приведен в Таблица 27.85.

Таблица 27.85. Формат регистра Transmit_Timestamp_Value_Lower_Register

Номер бита	Условное обозначение	Назначение
31:0	Transmit_Timestamp_Lower	Это поле содержит младшие 32 бита значения передающейся метки времени. ПО должно сначала прочитать этот регистр, прежде чем читать Transmit_Timestamp_Value_Upper_Register.

27.3.85 Регистр Transmit_Timestamp_Value_Upper_Register

Transmit_Timestamp_Registers используются для считывания текущей метки времени для указанного пакета передачи. Логика не будет обновлять это значение (не выполняет таймстемпинг для другого пакета), пока эти регистры не будут прочитаны. Регистры имеют RO доступ. Первый регистр содержит нижние 32 бита значения таймстемпа передачи, второй регистр содержит верхние 32 бита таймстемпа передачи.

Формат регистра Transmit_Timestamp_Value_Upper_Register приведен в Таблица 27.86.

Таблица 27.86. Формат регистра Transmit_Timestamp_Value_Upper_Register

Номер бита	Условное обозначение	Назначение
31:0	Transmit_Timestamp_Upper	Это поле содержит старшие 32 бита значения передающейся метки времени. ПО перед чтением этого регистр, должно прочитать Transmit_Timestamp_Lower_Upper_Register.

27.3.86 Регистр Receive_Timestamp_Value_Lower_Register

Receive_Timestamp_Registers используются для отображения текущей метки времени принятого пакета PTP, который соответствует запрограммированным параметрам (MsgID, L2 Ethertype или PTP UDP port и т.д.). Логика не будет обновлять это значение (не выполняет таймстемпинг другого пакета), пока эти регистры не будут прочитаны. Регистры имеют RO доступ. Первый регистр содержит нижние 32 бита значения принятой метки времени, второй содержит верхние 32 бита принятой метки времени.

Формат регистра Receive_Timestamp_Value_Lower_Register приведен в Таблица 27.87.

Таблица 27.87. Формат регистра Receive_Timestamp_Value_Lower_Register

Номер бита	Условное обозначение	Назначение
31:0	Receive_Timestamp_Lower	Это поле содержит младшие 32 бита значения принятой метки времени. ПО должно сначала прочитать этот регистр, затем Receive_Timestamp_Value_Upper_Register

27.3.87 Регистр Receive_Timestamp_Value_Upper_Register

Receive_Timestamp_Registers используются для отображения текущей метки времени принятого пакета PTP, который соответствует запрограммированным параметрам (MsgID, L2 Ethertype или PTP UDP port и т.д.). Логика не будет обновлять это значение (не выполняет таймстемпинг другого пакета), пока эти регистры не будут прочитаны. Регистры имеют RO доступ. Первый регистр содержит нижние 32 бита значения принятой метки времени, второй содержит верхние 32 бита принятой метки времени.

Формат регистра Receive_Timestamp_Value_Upper_Register приведен в Таблица 27.88.

Таблица 27.88. Формат регистра Receive_Timestamp_Value_Upper_Register

Номер бита	Условное обозначение	Назначение
31:0	Receive_Timestamp_Upper	Это поле содержит старшие 32 бита значения принятой метки времени. ПО перед чтением этого регистр, должно прочитать Receive_Timestamp_Lower_Upper_Register.

27.3.88 Регистр Receive_PTP_Packet_Attributes_Lower_Register

Receive_PTP_Packet_Attributes_Registers используются для отображения SrcID и SeqID полученного сообщения PTP, для которого была выполнена операция тайстемпинга. SrcID располагаются в 20 – 29 байтах PTP сообщения.

Формат регистра Receive_PTP_Packet_Attributes_Lower_Register приведен в Таблица 27.89.

Таблица 27.89. Формат регистра Receive_PTP_Packet_Attributes_Lower_Register

Номер бита	Условное обозначение	Назначение
31:16	SrcID_Lower	Это поле содержит младшие 16 бит SrcID, извлеченные из полученного PTP-пакета с меткой времени. Это байтовое смещение 28-29 в PTP пакете.
15:0	SeqID	Это поле содержит 16-битный SeqID, извлеченный из PTP-пакета приема с таймстемпингом.

27.3.89 Регистр Receive_PTP_Packet_Attributes_Middle_Register

Receive_PTP_Packet_Attributes_Registers используются для отображения SrcID и SeqID полученного сообщения PTP, для которого была выполнена операция тайстемпинга. SrcID располагаются в 20 – 29 байтах PTP сообщения.

Формат регистра Receive_PTP_Packet_Attributes_Middle_Register приведен в Таблица 27.90.

Таблица 27.90. Формат регистра Receive_PTP_Packet_Attributes_Middle_Register

Номер бита	Условное обозначение	Назначение
31:0	SrcID_Middle	Это поле содержит средние 32 бит SrcID, извлеченные из полученного PTP-пакета с меткой времени. Это байтовое смещение 24-27 в PTP пакете.

27.3.90 Регистр Receive_PTP_Packet_Attributes_Upper_Register

Receive_PTP_Packet_Attributes_Registers используются для отображения SrcID и SeqID полученного сообщения PTP, для которого была выполнена операция тайстемпинга. SrcID располагаются в 20 – 29 байтах PTP сообщения.

Формат регистра Receive_PTP_Packet_Attributes_Upper_Register приведен в Таблица 27.91.

Таблица 27.91. Формат регистра Receive_PTP_Packet_Attributes_Upper_Register

Номер бита	Условное обозначение	Назначение
31:0	SrcID_upper	Это поле содержит старшие 32 бита SrcID, извлеченные из полученного PTP-пакета с меткой времени. Это байтовое смещение 20-23 в PTP пакете.

27.3.91 Регистр Module_1588_Interrupt_Register

1588_IRQ_Register предоставляет информацию об условиях приема и передачи, которая должна контролироваться ПО.

Формат регистра Module_1588_Interrupt_Register приведен в Таблица 27.92.

Таблица 27.92. Формат регистра Module_1588_Interrupt_Register

Номер бита	Условное обозначение	Назначение
31:2	-	Резерв
1	Receive_Timestamp_IRQ	Устанавливается, когда получен пакет PTP, (который соответствует запрограммированным параметрам), и помечен как «метка времени», и значение доступно для чтения в регистрах Receive_Timestamp_(Lower/Upper)_Registers. Бит сбрасывается путем записи 1.
0	Transmit_Timestamp_IRQ	Устанавливается, передающийся пакет был помечен как «метка времени», и значение доступно для чтения в регистрах Transmit_Timestamp_(Lower/Upper)_Registers. Бит сбрасывается путем записи 1.

27.3.92 Регистр Module_1588_Interrupt_Enable_Register

1588_Interrupt_Enable_Register используется для маскирования прерываний от 1588 модуля.

Формат регистра Module_1588_Interrupt_Enable_Register приведен в Таблица 27.93.

Таблица 27.93. Формат регистра Module_1588_Interrupt_Enable_Register

Номер бита	Условное обозначение	Назначение
31:2	-	Резерв
1	Receive_Timestamp_Interrupt_Enable	Маска запроса на прерывание Receive_Timestamp_IRQ 1- прерывание разрешено 0- прерывание запрещено
0	Transmit_Timestamp_Valid_Interrupt_Enable	Маска запроса на прерывание Transmit_Timestamp_IRQ 1- прерывание разрешено 0- прерывание запрещено

27.4 Программная модель

GEMAC ядро пересылает пакеты, принятые по RGMII интерфейсу, в буфер приема, находящийся во внешней памяти, а также пересылает пакеты, отправляемые по RGMII интерфейсу, из буфера передачи, находящегося во внешней памяти. Дескрипторы,

хранящиеся по внешней памяти, содержат указатели на эти буферы. Receive и Transmit FIFO, входящие в состав контролера, служат для временного хранения передающихся и принимающихся кадров.

Используются два списка дескрипторов, один для передачи, и один для приема. Базовый адрес каждого списка хранится в регистре «*DMA Base Address Registers*». Список дескрипторов связан (явно или неявно). Для того чтобы создать кольцевую структуру списка, последний дескриптор должен указывать на первый. Явное формирование цепочек дескрипторов достигается установкой бит «*Second Address Chained*» в дескрипторах передачи и приема. Список дескрипторов постоянно находится во внешней памяти. Каждый дескриптор может указывать максимум на два буфера. Это дает возможность использования двух буферов по двум различным физическим адресам.

Буфер данных может содержать часть пакета, пакет целиком, но не может превышать размер пакета. Буфер содержит только данные. Вся информация о буфере содержится в дескрипторе. Для пакетов, расположенных в нескольких буферах формируются цепочки данных. Формирование цепочек данных можно разрешить или заблокировать. Буферы данных постоянно находятся во внешней памяти.

Рисунок 27.2 показывает пример кольцевой структуры дескрипторов. Здесь каждый дескриптор поддерживает до двух буферов, и следующий дескриптор следует за текущим. Последний дескриптор может указывать на первый.

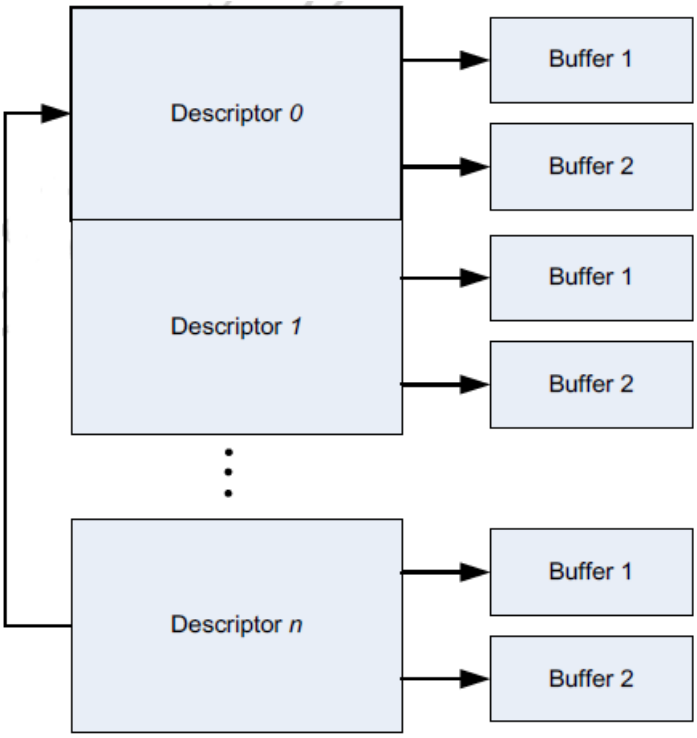


Рисунок 27.2. Кольцевая структура дескриптора

На Рисунок 27.3 показан пример цепочки дескрипторов. Здесь каждый дескриптор указывает только на один буфер. Адрес следующего дескриптора будет частью текущего дескриптора.

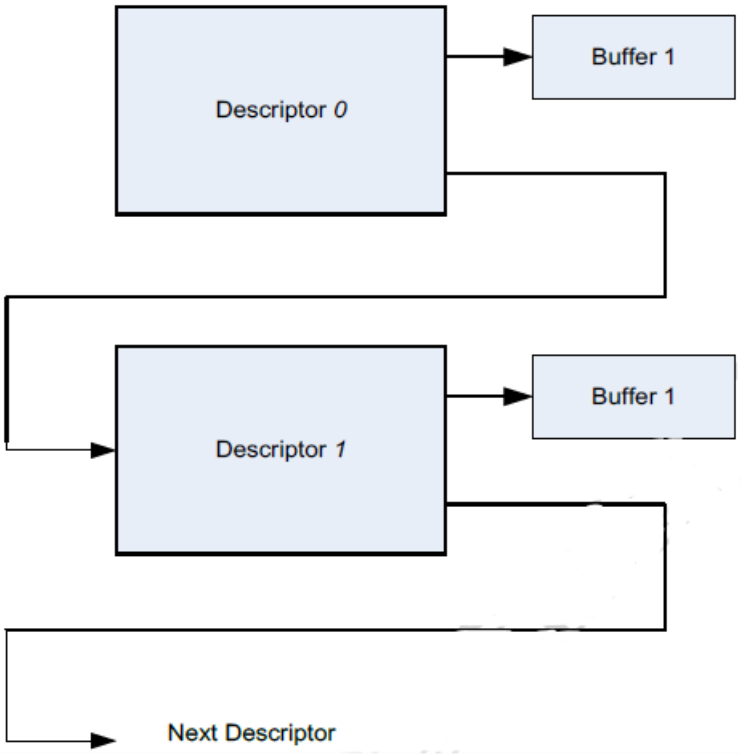


Рисунок 27.3. Цепочка дескрипторов

27.4.1 Дескрипторы приема

На Рисунок 27.4 представлена структура дескриптора приема в 32-х битном DWORD формате. Дескрипторы и буферы приема должны быть выровнены по 32-разрядной или 64-разрядной границе, в соответствии с конфигурацией DMA. Поддержка двух буферов, двух счетчиков байтов и двух указателей адреса в каждом дескрипторе позволяет порту быть совместимым с различными типами схем распредления памяти.

RDES0	OWN	Status		
RDES1	Control		Byte-Count. Buffer 2	Byte-Count Buffer 1
RDES2	Buffer Address 1			
RDES3	Buffer Address 2			

Рисунок 27.4. Структура дескриптора приема

27.4.1.1 Receive Descriptor 0 (RDES0)

RDES0 (Таблица 27.94. Receive Descriptor 0 (RDES0)) содержит статус принятого пакета, включая его длину, монопольную информацию дескриптора, а также разделители пакета.

Таблица 27.94. Receive Descriptor 0 (RDES0)

Номер разряда	Условное обозначение	Назначение
31	Own Bit	Признак готовности дескриптора. 1'b1: дескриптор готов к использованию 1'b0: дескриптор не готов DMA сбрасывает этот бит после приема пакета или при заполнении соответствующего данному дескриптору буфера.
30	First Descriptor	Если установлен данный бит, то дескриптор содержит первый буфер пакета. Если размер первого буфера равен нулю, то начало пакета содержится во втором буфере. Если размер второго буфера равен нулю, то начало пакета содержится во втором дескрипторе.
29	Last Descriptor	Если установлен данный бит, то дескриптор указывает на последний буфер пакета. В зависимости от длины пакета второй буфер может содержать данные или быть пустым.
28:14	Application Status	Это поле используется для хранения статуса пакета при условии, что установлен бит Last Descriptor.
13:0	Frame/Packet Length	Длина пакета (в байтах) передаваемого из Receive FIFO в память. Значение данного поля достоверно только тогда, когда установлен бит Last Descriptor.

27.4.1.2 Receive Descriptor 1 (RDES1)

RDES1 (Таблица 27.95) содержит управляющую информацию о структуре дескриптора, а также длину буферов #1 и #2.

Таблица 27.95. Receive Descriptor 1 (RDES1)

Номер разряда	Условное обозначение	Назначение
31	PTP_Packet	Хост должен установить этот бит в ноль при создании дескриптора. Контроллер GEMAC установит этот бит в единицу, если текущий принимаемый пакет является пакетом PTP, который соответствует запрограммированным параметрам PTP (L2 Ethertype или UDP Port number). Если MessageID (для пакетов PTP V2) или поле управления (для пакетов PTP V1) также совпадают, то также устанавливается бит метки времени PTP. В противном случае этот бит будет сброшен в ноль.
30	PTP_Timestamped	Хост должен установить этот бит в ноль при создании дескриптора. Контроллер GEMAC установит этот бит в единицу, если текущий принимаемый пакет является пакетом PTP, который соответствует запрограммированным параметрам PTP (L2 Ethertype или UDP Port number), а также соответствует запрограммированному MessageID или полю управления. Отметка времени для пакета, PTP SeqID и SrcID сохраняются в программно доступных регистрах. В противном случае этот бит будет сброшен в ноль.
29:27	-	Резерв
26	End of the Ring	Установка данного бита указывает о достижении последнего дескриптора в списке. DMA возвращается к базовому адресу кольцевого списка дескрипторов.
25	Second Address Chained	Если установлен данный бит, то второй адрес дескриптора указывает не на второй буфер, а на следующий дескриптор. Этот бит используется при создании цепочек дескрипторов.
24	-	Резерв
23:12	Buffer 2 Size	Размер в байтах второго буфера данных. Если значение этого поля равно нулю, то DMA игнорирует этот буфер и выбирает следующий дескриптор. Значение размера буфера должно лежать в диапазоне от 0 до 4095 байт. Это поле недостоверно, если установлен бит Second Address Chained.

Номер разряда	Условное обозначение	Назначение
11:0	Buffer 1 Size	Размер в байтах первого буфера данных. Если значение этого поля равно нулю, то DMA игнорирует этот буфер и использует буфер 2. Значение размера буфера должно лежать в диапазоне от 0 до 4095 байт.

27.4.1.3 Receive Descriptor 2 (RDES2)

RDES2 (Таблица 27.96) содержит указатель (физический адрес) на буфер #1

Таблица 27.96. Receive Descriptor 2 (RDES2)

Номер разряда	Условное обозначение	Назначение
31:0	Buffer Address 1	Отображает физический адрес первого буфера данных. На выравнивание начального адреса буфера не накладывается никаких ограничений.

27.4.1.4 Receive Descriptor 3 (RDES3)

RDES3 (Таблица 27.97) содержит указатель (физический адрес) на буфер #2 или, если используется структура списка связей, содержит указатель (физический адрес) следующего дескриптора.

Таблица 27.97. Receive Descriptor 3 (RDES3)

Номер разряда	Условное обозначение	Назначение
31:0	Buffer Address 2	Отображает физический адрес второго буфера данных. На выравнивание начального адреса буфера не накладывается никаких ограничений. Если установлен бит Second Address Chained, то данное поле содержит физический адрес следующего дескриптора. В зависимости о режима работы DMA адрес должен быть выровнен по 32-разрядной или 64-разрядной границе.

27.4.2 Дескрипторы передачи.

На Рисунок 27.5 представлена структура дескриптора передачи в 32-х битном DWORD формате. Дескрипторы и буферы передачи должны быть выровнены по 32-разрядной или 64-разрядной границе, в соответствии с конфигурацией DMA. Поддержка двух буферов, двух счетчиков байтов и двух указателей адреса в каждом дескрипторе позволяет порту быть совместимым с различными типами схем распределения памяти.

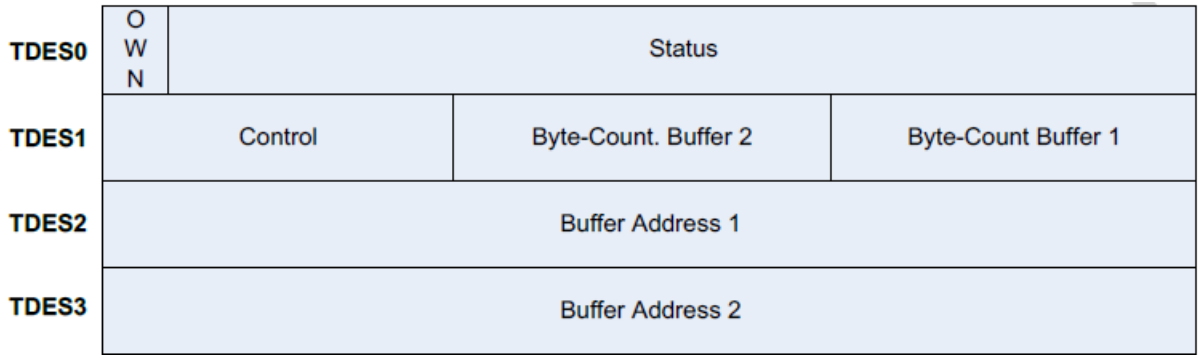


Рисунок 27.5. Структура дескриптора передачи

27.4.2.1 Transmit Descriptor 0 (TDES0)

TDES0 (Таблица 27.98) содержит статус передающегося пакета и монопольную информацию дескриптора.

Таблица 27.98. Transmit Descriptor 0 (TDES0)

Номер разряда	Условное обозначение	Назначение
31	Own Bit	Признак готовности дескриптора. 1'b1: дескриптор готов к использованию 1'b0: дескриптор не готов DMA сбрасывает этот бит после передачи пакета, или при опустошении соответствующего данному дескриптору буфера.
30	Timestam Packet	Установка этого бита указывает, что передаваемый пакет должен быть помечен меткой времени. Бит должен быть действительным, когда установлен первый сегмент. Этот бит может быть установлен только при включенной операции 1588 PTP. В противном случае этот бит игнорируется.
29:0	Transmit Frame/Packet Status	Это поле используется для хранения статуса пакета, при условии, что установлен бит Last Segment.

27.4.2.2 Transmit Descriptor 1 (TDES1)

TDES1 (Таблица 27.99) содержит управляющую информацию дескриптора, разделителей пакета, а также длину буферов #1 и #2.

Таблица 27.99. Transmit Descriptor 1 (TDES1)

Номер разряда	Условное обозначение	Назначение
31	Interrupt On Completion	Если данный бит установлен, то DMA после передачи пакета выставляет запрос на прерывание. Этот бит достоверен, если установлен бит Last Segment.
30	Last Segment	Если данный бит установлен, то буфер содержит последний сегмент пакета.
29	First Segment	Если данный бит установлен, то буфер содержит первый сегмент пакета.
28	Add CRC Disable	Если данный бит установлен, то вычисление и вставка поля FCS не производится. Если бит сброшен, то FCS вычисляется и вставляется в конец пакета. Этот бит достоверен, если установлен бит First Segment.
27	Disable Padding	Установка данного бита отключает добавление поля PAD к пакетам, размер которых менее 64 байт. Если бит сброшен, то в пакет добавляется поле PAD до

Номер разряда	Условное обозначение	Назначение
		значения minFrameSize (64 байта). Этот бит достоверен, если установлен бит First Segment.
26	End of the Ring	Установка данного бита указывает о достижении последнего дескриптора в списке. DMA возвращается к базовому адресу кольцевого списка дескрипторов.
25	Second Address Chained	Если установлен данный бит, то второй адрес дескриптора указывает не на второй буфер, а на следующий дескриптор. Этот бит используется при создании цепочек дескрипторов
24	Force EOP Error	Если данный бит установлен, то DMA контроллер формирует EOP с ошибкой к FIFO интерфейсу. Это используется для удаления сформированного пакета. Этот бит достоверен, если установлен бит Last Segment.
23:12	Buffer 2 Size	Размер в байтах второго буфера данных. Если значение этого поля равно нулю, то DMA игнорирует этот буфер и выбирает следующий дескриптор. Значение размера буфера должно лежать в диапазоне от 0 до 4095 байт. Это поле недостоверно, если установлен бит Second Address Chained.
11:0	Buffer 1 Size	Размер в байтах первого буфера данных. Если значение этого поля равно нулю, то DMA игнорирует этот буфер и использует буфер 2. Значение размера буфера должно лежать в диапазоне от 0 до 4095 байт.

27.4.2.3 Transmit Descriptor 2 (TDES2)

TDES2 (Таблица 27.100) содержит указатель (физический адрес) на буфер #1

Таблица 27.100. Transmit Descriptor 2 (TDES2)

Номер разряда	Условное обозначение	Назначение
31:0	Buffer Address 1	Отображает физический адрес первого буера данных. На выравнивание начального адреса буфера, не накладывается никаких ограничений.

27.4.2.4 Transmit Descriptor 3 (TDES3)

TDES3 (Таблица 27.101) содержит указатель (физический адрес) на буфер #2 или, если используется структура списка связей, содержит указатель (физический адрес) следующего дескриптора.

Таблица 27.101. Transmit Descriptor 3 (TDES3)

Номер разряда	Условное обозначение	Назначение
31:0	Buffer Address 2	Отображает физический адрес второго буфера данных. На выравнивание начального адреса буфера не накладывается никаких ограничений. Если установлен бит Second Address Chained, то данное поле содержит физический адрес следующего дескриптора. В зависимости о режима работы DMA адрес должен быть выровнен по 32-разрядной или 64-разрядной границе.

27.4.3 Формат дескриптора в памяти

Поскольку дескрипторы – структурированные данные, они должны формироваться в памяти в определенном порядке. Для увеличения гибкости GEMAC ядро поддерживает различные форматы представления дескриптора в памяти. *Descriptor Byte Order* (13-й и 19-й биты DMA Configuration Register) используются для выбора конфигурации контролера в

зависимости от принципа формирования дескриптора. Контроллер автоматически перестроит прочитанные данные дескриптора, как показано на Рисунок 27.2 и Рисунок 27.3 для дескрипторов передачи и приема соответственно.

27.4.3.1 Формат дескриптора в 32-разрядном режиме работы DMA

Если DMA работает в 32-разрядном режиме, то начало структуры дескрипторов должно быть выровнено по 4-байтной (DWORD) границе.

На Рисунок 27.6 представлено обычное расположение дескриптора, при котором GEMAC ядро ожидает, что в памяти MSB расположен слева, LSB справа. Дескриптор 32-разрядный, соответственно, адрес увеличивается со смещением на 4 байта.

TDES0[31:0]
TDES1[31:0]
TDES2[31:0]
TDES3[31:0]

Рисунок 27.6. Обычное расположение дескриптора (x32 режим)

TDES0[7:0]	TDES0[15:8]	TDES0[23:16]	TDES0[31:24]
TDES1[7:0]	TDES1[15:8]	TDES1[23:16]	TDES1[31:24]
TDES2[7:0]	TDES2[15:8]	TDES2[23:16]	TDES2[31:24]
TDES3[7:0]	TDES3[15:8]	TDES3[23:16]	TDES3[31:24]

Рисунок 27.7. Расположение дескриптора при установленном бите Descriptor Byte Order (x32 режим)

Если Descriptor Byte Order (бит 13 DMA Configuration Register) равен «1», то дескриптор располагается в памяти как показано на Рисунок 27.7.

27.4.3.2 Формат дескриптора в 64-разрядном режиме работы DMA

Если DMA работает в 64-разрядном режиме, то начало структуры дескрипторов должно быть выровнено по 8-байтной (QWORD) границе.

На 27.7 представлено обычное расположение дескриптора, при котором GEMAC ядро ожидает, что в памяти MSB расположен слева, LSB справа. Дескриптор 64-разрядный, соответственно, адрес увеличивается со смещением на 8 байт.

На рисунках ниже представлены варианты размещения дескрипторов в зависимости от значения битов «Descriptor Byte Order» (13-й и 19-й биты DMA Configuration Register)

TDES1[31:0]	TDES0[31:0]
TDES3[31:0]	TDES2[31:0]

Рисунок 27.8. Descriptor Byte Order = 2’b00

TDES1[7:0]	TDES1[15:8]	TDES1[23:16]	TDES1[31:24]	TDES0[7:0]	TDES0[15:8]	TDES0[23:16]	TDES0[31:24]
TDES3[7:0]	TDES3[15:8]	TDES3[23:16]	TDES3[31:24]	TDES2[7:0]	TDES2[15:8]	TDES2[23:16]	TDES2[31:24]

Рисунок 27.9. Descriptor Byte Order = 2’b01

TDES0[7:0]	TDES0[15:8]	TDES0[23:16]	TDES0[31:24]	TDES1[7:0]	TDES1[15:8]	TDES1[23:16]	TDES1[31:24]
TDES2[7:0]	TDES2[15:8]	TDES2[23:16]	TDES2[31:24]	TDES3[7:0]	TDES3[15:8]	TDES3[23:16]	TDES3[31:24]

Рисунок 27.10. Descriptor Byte Order = 2’b10

TDES0[7:0]	TDES0[15:8]	TDES0[23:16]	TDES0[31:24]	TDES1[7:0]	TDES1[15:8]	TDES1[23:16]	TDES1[31:24]
TDES2[7:0]	TDES2[15:8]	TDES2[23:16]	TDES2[31:24]	TDES3[7:0]	TDES3[15:8]	TDES3[23:16]	TDES3[31:24]

Рисунок 27.11. Descriptor Byte Order = 2’b11

27.4.4 Порядок расположения данных в памяти

Для выбора принципа хранения данных в буфере используется *Big/Little Endian Bit* (14-й и 20-й биты DMA Configuration Register).

В текущей реализации контроллера возможен только Little Endian формат. Начало расположения буфера в памяти можно не выравнивать. Рисунок 27.12 представлен Little Endian формат расположения данных для 32-разрядного режима работы DMA. Byte0 – первым передается в канал Ethernet, далее следуют Byte1, Byte2, Byte3 … ByteN.

Byte3	Byte2	Byte1	Byte0
Byte7	Byte6	Byte5	Byte4
Byte11	Byte10	Byte9	Byte8
Byte15	Byte14	Byte13	Byte12
….	….	….	….
ByteN	ByteN-1	ByteN-2	ByteN-3

Рисунок 27.12. Little Endian формат буфера (x32 режим)

Рисунок 27.13 представлен Little Endian формат расположения данных для 64-разрядного режима работы DMA. Byte0 – первым передается в канал Ethernet, далее следуют Byte1, Byte2, Byte3 … ByteN.

Byte7	Byte6	Byte5	Byte4	Byte3	Byte2	Byte1	Byte0
Byte15	Byte14	Byte13	Byte12	Byte11	Byte10	Byte9	Byte8
….	….	….	….	….	….	….	….
ByteN	ByteN-1	ByteN-2	ByteN-3	ByteN-4	ByteN-5	ByteN-6	ByteN-7

Рисунок 27.13. Big Endian формат буфера

27.5 Функциональное описание GEMAC

GEMAC ядро полностью поддерживает выполнение стандарта IEEE 802.3 на MAC уровне. MAC выполняет Full-Duplex и Half-Duplex операции при работе в режиме 10/100 Mbps и выполняет Full-Duplex операции при работе в режиме 1000 Mbps. MAC поддерживает 802.3 и Ethernet формат пакетов. Передающиеся данные пакета формируются в соответствии с IEEE 802.3 пункт 24. Аналогично, принятые данные распаковываются в соответствии с IEEE 802.3 пункт 24.

Поддерживается работа с Ethernet пакетами длиной от 64 до 1518 байт, не считая преамбулу и стартовый разделитель пакета. Ethernet пакет состоит из следующих частей:

- Preamble

- Start frame delimiter (SFD)
- Two address fields
- Type or length field
- Data field
- Frame check sequence (CRC value)

На Рисунок 27.14 представлен формат Ethernet пакета (передается слева направо).



Рисунок 27.14. Формат Ethernet пакета

В Таблица 27.102 отображено описание полей Ethernet пакета.

Таблица 27.102. Описание полей Ethernet пакета

Поле	Назначение
PREAMBLE	7-байтовое поле из последовательно меняющих друг друга нулей и единиц. В преамбуле байт представлен в виде 8'h55 (8'b10101010), младший бит принимается и передается первым.
SFD	Однобайтовое поле, обозначающее начало пакета 8'hD5 (8'b11010101). Младший бит принимается и передается первым.
DA	Шестибайтовое поле, содержащее индивидуальный, широковещательный или групповой адрес станции к которой направлен пакет.
SA	Шестибайтовое поле, содержащее индивидуальный адрес станции в которой сформировался пакет.
T/L	Двухбайтовое поле (Type/Length) определяющее формат кадра. Если значение поля больше или равно 16'h0600, то поле интерпретируется как поле Type, которое определяет тип протокола фрейма. Такой пакет называется Ethernet Type. Если значение поля меньше или равно 16'h05EE, то поле интерпретируется как поле Length, которое определяет количество байт во фрейме. Такой пакет называется 802.3.
DATA	Поле дата содержит от 46 до 1500 байт данных. Если данных меньше чем 46 байт, то они дополняются до 46 байт полем PAD.
FCS	Поле содержит результат (подсчитанный для полей DA, SA, T/L, DATA) 32-битной циклической проверки (CRC) передаваемого кадра. Поле FCS добавляется ко всем передаваемым кадрам и используется приемником для определения достоверности принятого пакета.

GEMAC ядро поддерживает Jumbo пакеты с нестандартной длиной до 9600 байт.

27.5.1 Передача пакета

В следующей секции детально описывается процесс передачи пакетов, который состоит из настройки дескриптора, буфера, настройки GEMAC ядра и, наконец, механизма передачи фрейма в RGMII интерфейс.

27.5.1.1 Подготовка пакета

CPU подготавливает передаваемый пакет, сохраняя полное содержимое фрейма (DA, SA, T/L, DATA и, опционально, FCS) в одном или нескольких буферах. Буфер может начинаться и заканчиваться с любой байтовой границы. Также готовится и размещается в памяти дескриптор передачи, соответствующий данному буферу, после чего подается команда GEMAC для загрузки дескриптора. Каждому передающемуся пакету может соответствовать один или несколько дескрипторов передачи. После загрузки дескриптора передачи, контроллер выбирает данные из внешней памяти и передает их во внутреннее Transmit FIFO. Формирование кадра начинается, когда количество данных в Transmit FIFO достигает запрограммированной границы, или, когда в Transmit FIFO загружен кадр полностью. Выбор границы заполнения FIFO основывается на скорости Ethernet операций и пропускной способности внутренней шины. Чем ниже скорость работы интерфейса Ethernet, тем ниже устанавливается граница заполнения FIFO. Грамотно выбранная граница заполнения Transmit FIFO предотвращает его переполнение. Формированием передающегося фрейма занимается расположенная в GEMAC ядре FSM, которая задерживает фактическую передачу данных на RGMII до тех пор, пока не выполнится условие минимального промежутка между кадрами Inter Frame Gap (IFG).

27.5.1.2 Формирование пакета

Формирование передаваемого фрейма подразумевает добавление 56-ти бит преамбулы, 8-ми бит SFD признака старта пакета (перед полем DA) и, опционально, 4-х байт FCS (после поля DATA в конце фрейма). Чтобы соблюсти ограничение minFrameSize (64 байта), поле DATA дополняется полем PAD (нулевыми данными). Дополнение полем PAD происходит, если поле DATA меньше 46 байт и дополнение не отключено в первом дескрипторе. CRC вычисляется согласно алгоритму, описанному в спецификации IEEE 802.3, с первого байта поля DA до последнего байта поля DATA, включая поле PAD (если оно было добавлено). Подсчитанная CRC добавляется в FCS поле передаваемого кадра. Можно отключить вставку FCS и подсчет CRC установкой бита Disable FCS Insertion в первом дескрипторе. Данная функция может быть полезна в том случае, когда CRC уже подсчитана и располагается вместе с кадром в памяти. В этом режиме к передаваемому кадру добавляется только преамбула и SFD.

27.5.1.3 Задержка фрейма, столкновение и повторная передача

Задержка передачи, столкновение фреймов и повторная передача возможны только при работе в Half-Duplex режиме. Эти функции реализованы в соответствии с протоколом CSMA/CD определенном в спецификации IEEE 802.3. В режиме Full-Duplex операции приема и передачи независимы, контроллер следит лишь за плотностью кадров (IFG).

В Half-Duplex режиме GEMAC ядро постоянно контролирует RGMII интерфейс. Передача кадра начинается только после простоя шины в 96-bit times (12 периодов тактовой частоты

RGMII). В случае повторной передачи после столкновения к 96-bit times добавляется требуемая пауза.

IFG разбит на две части - IFG1 и IFG2:

- IFG1 time (60-bit time): GEMAC контролирует время простоя RGMII интерфейса. Если обнаружена передача (установлен CRS сигнал), и в это время фрейм ожидает передачу, то передача откладывается до освобождения интерфейса, и отсчет IFG начинается заново.

- IFG2 time (36-bit time): Если обнаружена передача (установлен CRS сигнал), и в это время фрейм ожидает передачу, то GEMAC продолжает подсчет IFG. Передача начинается, когда IFG достигает значения 96-bit times, что ведет к столкновению кадров. Это обеспечивает равномерное использование интерфейса Ethernet всеми станциями.

Столкновение происходит, когда на Ethernet интерфейс одновременно выходит два или более Ethernet устройств. Когда GEMAC во время передачи обнаруживает столкновение, передача фрейма прекращается, и вместо кадра передается 32-разрядное JAM сообщение, состоящее из шестнадцатеричного 55555555. После передачи JAM сообщения начинается отсчет периода возврата (backoff wait period). Если коллизия обнаружена во время передачи преамбулы, то GEMAC ядро заканчивает передачу преамбулы, SFD, а лишь затем передает JAM сообщение. Это обеспечивает длину передаваемого пакета не менее 96 бит, как определено в спецификации. Повторная передача пакета выполняется после выдержки периода ожидания, основанного на двоичном экспоненциальном алгоритме. Задержка возврата является целочисленным множителем Slot Times (512 bit times). Количество Slot Times задержек перед n-ой попыткой передачи выбирается как случайное целое число r в диапазоне:

$$0 \leq r \leq 2^k$$

$$k = \min (n, N), \text{ где } N = 10$$

После 16-ти неудачных повторных попыток передачи GEMAC ядро прекращает передачу, сбрасывает передаваемый кадр из Transmit FIFO и инкрементирует счетчик чрезмерного количества коллизий.

27.5.1.4 Завершение пакета

Передача пакета может быть завершена нормально или с ошибкой. Кадр считается успешно переданным, если передан последний байт поля FCS. Нормальная передача заканчивается изменением статистических счетчиков передатчика. При ненормальных условиях кадр завершается с ошибкой. Различаются следующие ошибочные ситуации поддерживаемые ядром:

- Excessive Collision: В Half Duplex режиме передача заканчивается коллизией в 16 повторных передачах. Фрейм сбрасывается из Transmit FIFO. Счетчик Excessive Collision обновляется после смены дескриптора.

- Late Collision: Если в Half Duplex режиме столкновение обнаружено после нормального окна коллизии (64-байа) от преамбулы, то передача кадра прекращается. Оставшаяся часть кадра сбрасывается из Transmit FIFO. Счетчик Late Collision обновляется после смены дескриптора.

- Underflow: Если передающиеся данные не доступны во время передачи кадра, то передача заканчивается с ошибкой, CRC инвертируется. GEMAC ядро выставляет запрос на прерывание. Это случается, когда выбран очень низкий порог передачи, или в случае сильной загруженности шины AHB. Контроллер удалит недоступные на этапе передачи кадра данные сразу, как они будут догружены в Transmit FIFO.

- Jabber Length Exceeded: Если при передачи бессмысленных данных превышена длина Jabber Length, передача кадра прекращается, CRC инвертируется. GEMAC ядро выставляет запрос на прерывание. Оставшаяся часть кадра сбрасывается из Transmit FIFO.

После того как передача фрейма завершена (нормально или с ошибкой), и дескриптор отработан, ядро выставляет Transmit Interrupt и пытается выбрать следующий дескриптор. Если передатчик выключен во время передачи, то отключение происходит после завершения передачи текущего кадра.

27.5.2 Прием пакета

В следующей секции детально описывается процесс приема пакетов, который состоит из самого приема фрейма, анализа ошибок и, наконец, механизма передачи принятого фрейма в буфер приема.

27.5.2.1 Прием и распаковка кадра

GEMAC непрерывно контролирует RGMII интерфейс на наличие активности. Когда обнаружена активность, принимающая логика начинает обрабатывать данные преамбулы и SFD. Для правильной распаковки кадра GEMAC ядро должно принять от 0 до 7 байт преамбулы и SFD. Если при приеме преамбулы и SFD возникает ошибка, то прием пакета прерывается, а приемник перезапускается для ожидания следующего кадра.

Данные передающиеся после SFD (поле DA) принимаются и сохраняются в Receive FIFO. Данные преамбулы и SFD удаляются из полученного пакета до сохранения в FIFO. Если включена фильтрация адреса, содержимое поля DA сравнивается с запрограммированными адресами или с Multicast хеш-таблицей в случае Multicast пакета. Если результат

положительный, пакет принимается, в противном случае – удаляется из Receive FIFO. В следующем разделе описаны механизмы фильтрации адресов, поддерживаемые ядром.

Подсчет CRC начинается с первого байта поля DA и заканчивается последним байтом пакета, включая поле FCS. Если анализ CRC прошел успешно, то кадр принимается, в противном случае, если ядро не запрограммировано на прием кадров с ошибками, кадр отбрасывается. Контроль CRC можно отключить программно.

27.5.2.2 Завершение приема пакета

Прием пакета завершен успешно, если получен признак конца пакета по RGMII интерфейсу (снимается сигнал RX_DV). Далее описаны случаи аварийного приема фрейма:

- Receive FIFO Overflow: Receive FIFO переполняется, если DMA не успевает выгрузить поступающие в него данные. В этом случае принятый и стоящий в очереди на запись в FIFO кадр сбрасывается. Счетчик сброшенных кадров из-за переполнения FIFO инкрементируется.

- Jabber Length Exceeded: Если длина принимаемого кадра превышает значение запрограммированное в регистре Receive Jabber Length, то кадр обрезается, остаток полученного кадра игнорируется. Полученный обрезанный кадр может быть сброшен или передан в память с соответствующим битом ошибки.

- Frame Collision: Если при приеме произошло столкновение пакетов, то кадр рассматривается как фрагмент или как короткий (Runt) кадр и может быть сброшен или передан в память с соответствующим битом ошибки.

27.5.2.3 Передача пакета в память

DMA может работать в режиме Store-and-Forward или режиме Cut-Thru. В режиме Store-and-Forward пакет передается в память только в случае его полной загрузки в FIFO. В режиме Cut-Thru передача пакета в память начинается сразу, как только количество принятых в FIFO байт превысит значение, запрограммированное в регистре Packet Start Threshold. Содержимое пакета переносится во внешний буфер, адрес которого указан в дескрипторе. В зависимости от размера кадра и размера выделенного буфера, пакет может быть описан как одним, так и несколькими дескрипторами. Когда последний сегмент кадра передан в память, в поле RDES0 последнего дескриптора записывается соответствующий статус. Дополнительно возможна установка запроса на прерывание, информирующая о наличии в памяти нового принятого кадра.

При работе в режиме Store-and-Forward GEMAC ядро по умолчанию сбрасывает из Receive FIFO кадры с ошибкой и кадры, не прошедшие адресную фильтрацию. При работе в Cut-Thru режиме кадры с ошибками сбрасываются, только когда их длина не превышает значение Rx Packet Start Threshold, если же граница превышена, то пакет передается в

память с установкой бита ошибки. В режиме Cut-Thru значения параметра Rx Packet Start Threshold должно быть не менее 12 для обеспечения адресной фильтрации. Далее описаны варианты поведения ядра в различных ситуациях:

- Address Filter Fail: Возникает когда пакет с полем DA не проходит механизм адресной фильтрации.
    - В режиме Store-and-Forward пакет сбрасывается из Receive FIFO.
    - В режиме Cut-Thru пакет сбрасывается из Receive FIFO только в случае, когда значение Rx Packet Start Threshold больше или равно 12, иначе пакет передается во внешнюю память.
- FCS Error: Возникает, когда пакет не прошел CRC проверку, или предполагается наличие CRC ошибки.
    - В режиме Store-and-Forward пакет сбрасывается из Receive FIFO.
    - В режиме Cut-Thru пакет сбрасывается из Receive FIFO только в случае, когда его длина меньше значения Rx Packet Start Threshold.
- Runt Frame: Если длина кадра (от поля DA до FCS включительно) менее 64-х байт. Большинство Runt Frame возникает в результате коллизий.
    - В режиме Store-and-Forward пакет сбрасывается из Receive FIFO.
    - В режиме Cut-Thru пакет сбрасывается из Receive FIFO только в случае, когда его длина меньше значения Rx Packet Start Threshold.
- Receive Error: Возникает при обнаружении PHY уровнем ошибки декодирования бит (установлен RX_ER). В этом случае контроллер выставляет ошибку приема, а также ошибку CRC.
    - В режиме Store-and-Forward пакет сбрасывается из Receive FIFO.
    - В режиме Cut-Thru пакет сбрасывается из Receive FIFO только в случае, когда его длина меньше значения Rx Packet Start Threshold.
- Frame Too Long: Если длина пакета превышает запрограммированное значение Max Frame Length. Для приема Jumbo (огромных) пакетов необходимо соответствующим образом программировать регистр Max Frame Length.
    - В режиме Store-and-Forward пакет сбрасывается из Receive FIFO.
    - В режиме Cut-Thru пакет передается в память с установкой бита ошибки.

- Length Error: Возникает, когда в принимаемом пакете поле Type/Length указывает длину пола данных, а размер принятых данных не соответствует этому значению. Если данное несоответствие присутствует, то считается, что у кадра ошибка длины.
    - В режиме Store-and-Forward пакет сбрасывается из Receive FIFO.
    - В режиме Cut-Thru пакет сбрасывается из Receive FIFO только в случае, когда его длина меньше значения Rx Packet Start Threshold.
- Jabber Error: Если длина принимаемого кадра превышает запрограммированное значение Jabber Length, то его прием прекращается, и устанавливается Jabber Error.
    - В режиме Store-and-Forward пакет сбрасывается из Receive FIFO.
    - В режиме Cut-Thru обрезанный пакет передается в память с установкой бита ошибки.

В режиме Store-and-Forward контроллер может быть запрограммирован на передачу пакетов с ошибками или не прошедших фильтрацию в память с соответствующей установкой битов статуса. Далее описаны возможные варианты управления:

- Promiscuous Mode: При работе в этом режиме ядро передает в память пакеты, не прошедшие адресную фильтрацию вместо того, чтобы сбросить их из Receive FIFO.
- Pass Bad Frames: При работе в этом режиме ядро передает в память пакеты с ошибками вместо того, чтобы сбросить их из Receive FIFO. Соответствующий бит статуса устанавливается в последнем дескрипторе пакета.

Существует три случая, не связанных с ошибками, при возникновении которых кадр сбрасывается из Receive FIFO. Это происходит, если DMA не может передать в память принятый кадр. Далее описаны эти три случая:

- Receive DMA is in STOPPED state: Если принимающий DMA находится в состоянии STOPPED, и пакет принят по RGMII интерфейсу, то он будет сброшен из Receive FIFO. При каждом сбросе кадра по этой причине инкрементируется счетчик Stop Flush Counter.
- Receive Descriptor Unavailable: В случае, если кадр принят, а ядро не может получить дескриптор приема (например, если он еще не создан), то кадр сбрасывается из Receive FIFO. В этом случае инкрементируется Missed Frame Counter.
- Receive FIFO Full: Receive FIFO может быть заполнено, если DMA не может обеспечить нужную скорость передачи данных в память. В этом случае кадр сбрасывается из Receive FIFO. Если переполнение FIFO наступает в середине принимаемого кадра, то принятая часть сразу сбрасывается, а оставшаяся не

принимается. При возникновении данной ошибки инкрементируется счетчик Buffer Full Frame Drop Counter.

При работе ядра в Cut-Thru режиме кадр усекается, если начальная часть кадра принята и при этом переполнилось FIFO. В этом случае усеченная часть кадра передается в память с отметкой об ошибке FCS и скорректированной длиной поля Frame Length в соответствии с длиной усеченного кадра. При усечении кадра, вызванного переполнением буфера, инкрементируется счетчик Buffer Full Frame Truncated Counter.

Контроллер может быть запрограммирован на удаление поля FCS перед передачей кадра в память. Когда поле FCS удаляется, поле Frame Length в статусе кадра отображает длину пакета минус FCS.

27.5.2.4 Статус пакета

Если принятый кадр успешно передан в память, то статус кадра записывается в RDES0 последнего дескриптора (дескриптора, у которого установлен бит Last Descriptor). В Таблица 27.103 описаны поля регистра статуса кадра.

Таблица 27.103. Поля регистра статуса кадра

Номер разряда	Условное обозначение	Назначение
13:0	Frame/Packet Length	Отображает длину (в байтах) переданного в память пакета. Отображается количество переданных байт в память без отброшенных во время обработки пакета полей.
14	Alignment Error	Ошибка выравнивания пакета. Возникает при работе с MII интерфейсом, если принято неполное количество байт и возникла ошибка CRC.
15	Runt Frame	Информирует о том, что принятый пакет меньше минимального размера пакета (64 байта). Такая ситуация может возникнуть при возникновении коллизии.
16	Ethernet Type Frame	Информирует о том, что принятый пакет является пакетом типа Ethernet. Поле Type/Length имеет значение, которое больше или равно 0x0600. Если этот бит не установлен, то пакет относится к типу 802.3 и поле Type/Length отображает длину поля DATA.
17	VLAN Frame	Информирует о том, что пакет имеет один или более VLAN Tags.
18	Multicast Frame	Информирует о том, что поле DA имеет групповой адрес.
19	Broadcast Frame	Информирует о том, что поле DA имеет широковещательный адрес.
20	CRC Error	Информирует о том, что подсчитанный CRC пакета не соответствует ожидаемому.
21	Max. Frame Length Violation Error	Информирует о том, что размер принятого пакета превышает значение, запрограммированное в регистре Max. Frame Length Register.
22	Jabber Error	Информирует о том, что размер принятого пакета превышает значение, запрограммированное в регистре Jabber Register
23	Length Error	Информирует о том, что длина поля DATA не соответствует значению, запрограммированному в поле Type/Length. Это относится только к пакетам типа 802.3. Если длина поля DATA больше чем значение, указанное в поле Type/Length, то дополнительные байты считаются полем PAD и Length Error не возникает.

Номер разряда	Условное обозначение	Назначение
24	MAC Address#1 Match	Информирует о том, что поле DA принимаемого пакета совпало со значением, запрограммированным в регистре MAC Address#1.
25	MAC Address#2 Match	Информирует о том, что поле DA принимаемого пакета совпало со значением, запрограммированным в регистре MAC Address#2.
26	MAC Address#3 Match	Информирует о том, что поле DA принимаемого пакета совпало со значением, запрограммированным в регистре MAC Address#3.
27	MAC Address#4 Match	Информирует о том, что поле DA принимаемого пакета совпало со значением, запрограммированным в регистре MAC Address#4.
28	PAUSE Control Frame	Информирует о том, что текущий пакет является контролирующим пакетом PAUSE, и он предназначен данному устройству.

27.5.3 Фильтрация адреса

GEMAC ядро поддерживает фильтрацию адресов по полю DA входящего кадра. На основании типа DA (индивидуальный, групповой, широковещательный) используются различные схемы адресной фильтрации, по результатам которой кадр принимается или отклоняется. Если входящий пакет не проходит адресную фильтрацию, и не включен режим Promiscuous mode, то кадр не принимается в FIFO. Т.к. поле DA передается первым в пакете, то результат фильтрации адресов доступен уже после приема нескольких байт. Это помогает решить, нужно ли писать кадр в FIFO или нет.

Адресная фильтрация включена при работе контроллера в режиме Store-and-Forward, а также при работе в режиме Cut-Thru при условии, что значение регистра Rx Packet Start Threshold не менее 12.

В следующих разделах описано поведение ядра в зависимости от типа DA.

27.5.3.1 Пакеты с индивидуальным адресом (Unicast Frames)

GEMAC ядро может быть запрограммированно на использование до четырех MAC адресов с именами от MAC Address #1 до MAC Address #4. Регистр MAC Address Control используется для индивидуального включения и выключения MAC адресов. Если во входящем кадре поле DA совпадает с одним из четырех регистров MAC Address, и использование данного регистра разрешено (установлен соответствующий бит включения), то кадр принимается, иначе отбрасывается.

GEMAC ядро также поддерживает инверсную фильтрацию MAC адресов по указанным ранее регистрам MAC Address. Если установлен бит Inverse MAC Address, то для любого из четырех MAC адресов перед сравнением с полем DA выполняется операция инверсии. Если поле DA совпадает со значением регистра MAC Address, но включена инверсия, то фильтрация считается неуспешной.

27.5.3.2 Пакеты с групповым адресом (Multicast Frames)

Если первый бит поля DA равен «1», то адрес считается групповым. Для групповых адресов выполняется хеш-фильтрация на основании 64-разрядной групповой хеш-таблицы.

Для выполнения групповой хеш-фильтрации сохраняется результат CRC проверки принятого DA поля и младшие 6 бит сохраненного CRC используются в качестве индекса 64-разрядной групповой хеш-таблицы. Если по данному индексу в таблице хранится значение «1», то групповой DA адрес принимается, иначе – отклоняется.

Из-за малого размера хеш-таблицы схема фильтрации групповых адресов является не эффективной, что приводит к приему ненужных пакетов.

27.5.3.3 Пакета с широковещательным адресом (Broadcast Frames)

Если поле DA состоит из всех «1», то данный адрес и пакет является широковещательным. Все широковещательные пакеты принимаются GEMAC ядром и передаются во внешнюю память.

27.5.3.4 Разнородный режим (Promiscuous Mode)

Когда ядро работает в режиме Promiscuous Mode, принимаются все пакеты независимо от поля DA. В этом режиме все пакеты, принятые по RGMII интерфейсу, передаются во внешнюю память. Результат фильтрации адресов отмечается в поле статуса пакета (например, групповой пакет, широковещательный пакет, совпал Address #1 и т.д.)

27.5.4 Управление потоком данных

Управление потоком позволяет снизить вероятность переполнения буфера, соответственно, и снизить количество пропущенных пакетов, также позволяет локально контролировать уровень загрузки сети. Это может быть достигнуто путем отправки сообщения передающей станции о почти полном буфере на принимающей станции. Управление потоком одного участника передачи позволяет управлять потоком другого участника с возможностью игнорирования данных команд. Примером может служить не требующий ответа пакет PAUSE.

Когда GEMAC ядро работает в режиме Full-Duplex, поддерживаются Flow-Control операции приема и передачи независимо друг от друга. Предусмотрено отдельное включение/выключение поддержки операций Flow-Control как для приема, так и для передачи.

GEMAC ядро поддерживает набор регистров, программирование которых позволяет управлять пакетами Flow-Control PAUSE на RGMII интерфейсе. Прием пакета PAUSE осуществляется автономно, если данный режим включен.

27.5.4.1 Прием пакета MAC Flow-Control (PAUSE)

Если в GEMAC ядре включен прием и декодирование пакета PAUSE, то контроллер отслеживает все кадры с полем Length/Type равным Control Fame (16'h8808). Когда принят управляющий пакет, выполняется дополнительное декодирование для подтверждения, что принят действительно пакет PAUSE. Для положительного декодирования пакета PAUSE должны быть выполнены следующие условия:

1. Поле DA должно совпадать с одним из четырех запрограммированных MAC адресов или содержать групповой адрес 0x01_80_C2_00_00_01, указывающий, что кадр предназначен для данного устройства.

2. Поле Length/Type должно иметь значение 0x88_08 указывающее, что это управляющий кадр.

3. Поле OPCODE должно иметь значение 0x00_01 указывающее, что это управляющий кадр PAUSE.

4. Длина пакета должна быть равна 64 байтам.

5. Пакет не должен иметь ошибок приема и ошибок CRC.

Когда все вышеперечисленные условия выполнены, значение PAUSE Quanta извлекается из управляющего кадра PAUSE. Это значение используется для отключения передатчика до тех пор, пока не иссякнет счетчик паузы. Отключение передатчика блокирует отправку любых пакетов на MII/GMII интерфейс. Кванта определяет количество временных интервалов Slot time (512 bit times) для паузы. Если принят пакет PAUSE и значение кванты не нулевое, то в таймер загружается это значение и передача останавливается. Если во время остановки передается пакет, то остановка откладывается до окончания передачи. Передатчик отключается до тех пор, пока значение счетчика паузы не станет нулевым.

Если во время паузы приходит еще один управляющий кадр с квантой, равной нулю, то передатчик немедленно включается и выходит из режима PAUSE.

Передатчик формирует пакет PAUSE Flow-Control по полученной команде даже в том случае, если он выключен.

В GEMAC ядре можно отключить передачу принятого пакета PAUSE в память. Если установлен бит Block Pause Frame в регистре Flow Control, то управляющий пакет PAUSE сбрасывается из Receive FIFO. В нормальном режиме контроллер передает пакет PAUSE в память, как и любой другой пакет.

27.5.4.2 Формирование пакета MAC Flow-Control (PAUSE)

Контроллер может генерировать пакет PAUSE Flow-Control на интерфейс RGMII по программному запросу или при заполнении/опустошении Receive FIFO и пересечении верхнего/нижнего порогового значения. GEMAC ядро поддерживает формирование пакета PAUSE с различными запрограммированными параметрами. Следующие поля управляющего пакета PAUSE могут быть программно изменены:

1. **DA field:** Поле DA управляющего пакета PAUSE может иметь индивидуальный адрес, запрограммированный в регистре Flow-Control Destination Address или зарезервированный групповой адрес (0x01_80_C2_00_00_01).

2. **SA field:** Поле SA управляющего пакета PAUSE имеет индивидуальный адрес, запрограммированный в регистре Flow-Control Source Address.

3. **Type/Length field:** Для контролирующего пакета это поле постоянно (16'h88_08)

4. **OPCODE:** В поле OPCODE фиксируется значение 16’h00_01, которое говорит о том, что контролирующий пакет является контролирующим пакетом PAUSE.

5. **PAUSE Timer:** Значение, запрограммированное в регистре PAUSE Timer.

6. **PAD:** GEMAC ядро дополняет кадр нулевыми значениями пола PAD до minFrameSize (64 байта) включая поле FCS.

7. **FCS:** Контроллер вычисляет значение CRC для сгенерированного пакета и добавляет его в конец кадра.

Если поступила программная команда на формирование пакета PAUSE, то управляющий кадр собирается с полями по описанному выше принципу. Если в текущий момент времени передатчик передает пакет, то формирование и передача пакета PAUSE откладывается до завершения передачи с выдержкой времени IFG. Передача контролирующего пакета PAUSE осуществляется даже в том случае, если передатчик отключен ранее принятой командой управления потоком. До того, как программировать следующий кадр PAUSE нужно убедиться, что предыдущий сформирован и передан, статус передачи показывает бит DONE регистра Flow-Control Control Register.

Кроме того, если включено автоматическое формирование кадра PAUSE (Auto Pause Frame Generation) основанное на уровне заполнения Receive FIFO, то контроллер непрерывно следит за уровнем на предмет пересечения верхнего/нижнего порогового значения. Если уровень заполнения достигает верхнего порогового значения, то пакет PAUSE генерируется с величиной паузы запрограммированной в регистре Auto Hi Pause Time Register. После этого, если происходит опустошение Receive FIFO, и пересекается нижнее пороговое значение, генерируется пакет PAUSE с величиной паузы запрограммированной в регистре

Auto Low Pause Time Register. Таким образом, при многократном пересечении одного и того же порогового значения формируется только один пакет – при первом пересечении.

27.5.5 Управление приоритетом (Priority Flow-Control)

Если контроллер работает в полнодуплексном режиме, поддерживаются операции управления потоком как для приема, так и для передачи (до 8 приоритетов), как указано в спецификации IEEE 802.3Qbb. Программное обеспечение может включать/отключать операции управления потоком независимо и для передачи и для приема, а также для отдельных приоритетов.

В контроллере реализованы регистровы, позволяющие программно управлять генерацией кадров «Priority PAUSE Flow-Control» на интерфейсе RGMII. Прием и декодирование приоритетных кадров PAUSE Flow-Control выполняется автономно при включении. В следующих разделах описывается режим управления потоком GEMAC для передачи и приема.

27.5.5.1 Прием кадров Priority Flow-Control (PAUSE)

Если контроллеру разрешен прием и декодирование кадров Priority PAUSE Flow Control, то будут анализироваться все кадры поля Type/Length которых равны контрольному кадру (16’h8808). Когда кадр управления получен, выполняется дополнительное декодирование кадра управления, чтобы убедиться, что это действительный «Priority PAUSE Flow-Control», адресованный с зарезервированным многоадресным адресом. Для правильного декодирования PAUSE Flow-Control должны быть выполнены следующие условия:

1. поле DA должно совпадать с зарезервированным адресом многоадресной рассылки 0x01_80_C2_00_00_01, указывающим, что кадр адресован этому устройству.
2. поле Type/Length должно иметь значение 0x88_08, указывающее, что это управляющий кадр.
3. поле OPCODE должно иметь значение 0x01_01, указывающее управляющий кадр PAUSE.
4. длина кадра должна быть 64 байта.
5. кадр не должен иметь ошибки приема или ошибки CRC.

Когда все вышеперечисленные условия выполнены, вектор приоритета извлекается. Поле вектора времени для включенных приоритетов объединяются по «И» с загружнным вектором приоритета, и в логике использования этих результирующих приоритетов создается индикация XOFF. Контроллер начинает декремент этих счетчиков на основе

слота таймера. Когда эти таймеры достигают нуля или принимается новый кадр Priority PAUSE flow Control, который обнуляет счетчик, индикация XOFF сбрасывается.

В контроллере можно заблокировать передачу кадров Priority PAUSE в память хоста. Когда бит “Block Pause Frame” установлен в регистре “Flow Control Register”, кадры Priority PAUSE Control сбрасываются из принимаемого FIFO. В обычном режиме ядро GEMAC передает кадр Priority PAUSE Control в память хоста, как и любой другой кадр данных.

27.5.5.2 Формирование Priority Flow-Control (PAUSE) кадров

Контроллер может генерировать кадры “Priority PAUSE Flow-Control” на интерфейс RGMII под управлением программного обеспечения. GEMAC ядро поддерживает набор регистров, используемых для генерации фреймов “Priority PAUSE Control” с различными опциями. Ниже приведены поля кадра “Priority PAUSE Control”, некоторые из которых управляются программным обеспечением:

1. DA: всегда установлено на зарезервированный адрес многоадресной рассылки (0x01_80_C2_00_00_01).
2. SA: является адресом одноадресной передачи, который запрограммирован в регистрах “Flow-Control Source Address Registers”.
3. Type/Length: 16’h88_08
4. OPCODE: 16’h01_01
5. Priority Vector: вектор приоритета
6. Time VectorN: для каждого из приоритетов TimeVector устанавливается в 0xFFFF или 0x0000 в зависимости от того, указано или нет условие XOFF или условие XON
7. PAD: дополняется нулями, чтобы довести кадр до минимального размера (64 байта), включая FCS
8. FCS: контроллер рассчитывает CRC для сгенерированного кадра, и вставляет его в конец кадра

Когда программное обеспечение программирует контроллер для создания кадра «Priority PAUSE Control» или когда логика пользователя переключает один из 8 сигналов приоритета XOFF/XON, ядро собирает кадр “Priority PAUSE Control” со значениями полей, как указано выше. Если передатчик в текущий момент передает кадр, то передача кадра “Priority PAUSE Control” откладываетчя до завершения передачи текущего кадра с выдержкой времени IFG. “Priority PAUSE Control” фрейм передается, даже если передатчик отключен из-за более раннего приема кадра “Flow-Control”. Программное обеспечение должно постоянно опрашивать бит DONE в регистре “Flow-Control Control Register”, чтобы убедиться, что

передача кадра “Priority PAUSE” завершена, прежде чем программировать передачу другого кадра “Priority PAUSE”.

27.5.6 Управление формированием MDIO пакетов

GEMAC ядро поддерживает формирование управляющих пакетов для внешнего устройства PHY по интерфейсу MDC/MDIO. Данное управление позволяет читать/писать внутренние регистры внешнего PHY. Один MDC/MDIO интерфейс может поддерживать до 32-х PHY устройств.

MDC/MDIO интерфейс – это низкоскоростной интерфейс с максимальной частотой тактового сигнала 400 кГц. Ядро формирует синхросигнал MDC путем деления частоты axi_clk на 64. Также для формирования управляющих пакетов используются данные регистров MDIO Data Register и MDIO Control Register. MDIO Control Register содержит поле PHYAD/REGAD управляющего пакета, а также признак команды для выполнения (чтение или запись). MDIO Data Register содержит записываемые данные - во время записи, или прочитанные данные – после команды чтения.

Как только поступила команда на формирование управляющего кадра, контроллер собирает управляющий кадр в соответствии со спецификацией IEEE 802.3 и передает собранный кадр, последовательно сдвигая биты. Во время команды записи данные, содержащиеся в регистре MDIO Data Register, формируют поле DATA. При чтении данные, возвращенные внешним PHY в поле DATA, загружаются в регистр MDIO Data Register. На время передачи кадра в регистре MDIO Control Register устанавливается бит BUSY. Таким образом, перед подачей новой команды необходимо убедиться в завершении предыдущей.

27.5.7 IEEE 1588/PTP (двухступенчатый режим)

Контроллер GEMAC помогает протоколу IEEE-1588, предоставляя информацию о метках времени для специфичных входных и выходных пакетов под управлением ПО. Поддержка 1588 стандарта обеспечивается только при работе в дуплексном режиме на скоростях 10/100/1000 Мбит/с. В следующих разделах описывается процесс PTP синхронизации времени, функции выполняемые контроллером, а также функции, которые должны выполняться программным обеспечением для достижения полной функциональности протокола IEEE-1588.

27.5.7.1 Обзор PTP

Работа сети с поддержкой протокола точного времени (PTP) делится на два этапа: инициализация и синхронизация времени. На этапе инициализации каждый ведущий узел запускает отправку пакетов Announce, которые включают параметры синхронизации его часов. После получения пакета Announce узел сравнивает полученные параметры синхронизации со своими собственными параметрами, и если полученные параметры

лучше, то этот узел переходит в подчиненное состояние и прекращает отправку пакетов SYNC. Находясь в подчиненном состоянии, узел непрерывно сравнивает входящий пакет с выбранным на данный момент мастером, и, если новые параметры синхронизации лучше, то происходит переключение на часы этого мастера. В конце концов, лучшие мастер часы (master clock) выбраны. Каждый узел имеет определенный интервал времени ожидания, в течение которого, если от выбранных им главных часов не было получено ни одного пакета SYNC, он возвращается в состояние мастера и начинает отправку пакетов SYNC до тех пор, пока не будут выбраны новые лучшие ведущие часы (BMC).

Этап синхронизации времени отличается для главного и подчиненного узлов следующим образом:

• Master. Должен периодически отправлять пакет SYNC с метками времени по тракту передачи (как можно ближе к PHY). После пакета SYNC отправляется пакет FOLLOW_UP, который включает значение метки времени, сохраненной в пакете SYNC. Кроме того, мастер должен отметить временные метки пакетов DELAY_REQ на своем пути приема и вернуть подчиненному, который отправил ему значение временной метки, используя пакет DELAY_RESPONSE.

• Slave. Должен отмечать время каждого входящего пакета SYNC и, если он пришел от выбранного мастера, использует это значение для вычисления смещения. Кроме того, он должен периодически отправлять пакеты DELAY_REQ, чтобы рассчитать задержку пути от своего мастера. На каждый отправленный пакет DELAY_REQ накладывается метка времени и хранится. Используя значение, полученное в пакете DELAY_RESPONSE, ведомое устройство может рассчитать задержку пути от ведущего устройства к ведомому.

Маршрут протокола синхронизации и вычисления смещения показаны на Рисунок 27.15:

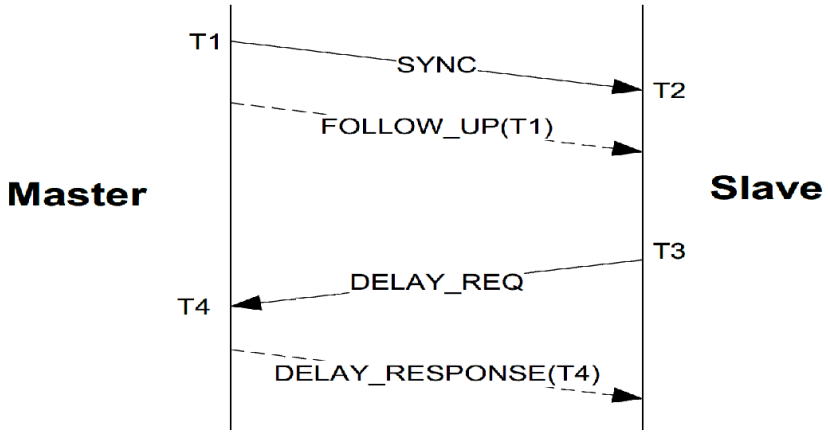


Рисунок 27.15. Протокол синхронизации

- T1 - метка времени передаваемого пакета SYNC в ведущем устройстве.
- T2 - значение метки времени полученного пакета SYNC в ведомом устройстве.

- T3 - метка времени передаваемого пакета DELAY_REQ в ведомом устройстве.
- T4 - метка времени полученного пакета DELAY_REQ в мастере.
- Пакет FOLLOW_UP содержит встроенное в него значение T1
- Пакет DELAY_RESPONSE содержит встроенное в него значение T4.

Смещение вычисляется в подчиненном устройстве следующим образом:

Toffset = [(T2-T1) - (T3-T4)] / 2

27.5.7.2 Распределение обязанностей контроллера и ПО

В вышеупомянутом потоке синхронизации времени контроллер GEMAC выполняет следующие функции:

- Определяет (из дескрипторов) TX пакеты требующих метки времени.
- Сохраняет значение передающейся метки времени для ПО.
- Анализирует входящие пакеты и идентифицирует PTP пакеты.
    - Поддерживает форматы кадров IEEE 1588 V1 и V2.
    - V1 PTP в полезной нагрузке UDP через IPV4
    - V2 PTP как специальные Ethertype или UDP через IPV4 или IPV6.
- Идентифицирует RX пакеты, требующие метки времени (на основе идентификатора сообщения (Message ID) PTP).
- Пропускает PTP-пакет и указатели с метками времени для ПО.
- Сохраняет значение принятых меток времени и SeqID/SrcID для ПО.
- Реализует системный таймер с программируемым значением приращения и периодом приращения.
- Предоставляет программный интерфейс для настройки системного таймера.

Ожидается, что ПО будет отвечать за следующие функции:

- Выполнение протокола BMC, что означает определение состояния узла (ведущего или подчиненного) и выбор главных часов, если они находятся в подчиненном состоянии.

- Генерация и анализ PTP-пакетов.
- Расчет смещения времени и непрерывная настройка системного времени с использованием программного интерфейса.

Таблица 27.104 отображает хронологические события потока синхронизации сверху, а также действия, выполняемые ядром GEMAC, и действия, которые должны выполняться программным обеспечением.

Таблица 27.104. Хронологические события потока синхронизации (двухступенчатый режим)

Действие	Обязанность
Генерация пакета SYNC с индикацией пакета метки времени (master)	ПО
Отметка времени пакета SYNC и предоставление значения отметки времени, T1 (Master)	Контроллер
Отметка времени входящего пакета SYNC, T2, и предоставление его вместе с деталями пакета PTP (SeqID / SrcID) и указателями отметки времени (slave)	Контроллер
Чтение значения метки времени T1 отправки пакета SYNC из регистров и включение ее в пакет FOLLOW_UP и передача пакета (master)	ПО
Как только пакет FOLLOW_UP получен, извлечение T1 и получение T2 из регистров (slave)	ПО
Генерация пакета DELAY_REQ с указанием пакета метки времени в дескрипторе передачи (slave)	ПО
Отметка времени пакета DELAY_REQ и предоставление значения отметки времени, T3 (Slave)	Контроллер
Отметка времени входящего пакета DELAY_REQ, T4 и предоставление его вместе с деталями пакета PT (SeqID/SrcID) и индикацией отметки времени (Master)	Контроллер
Чтение значения принятой метки времени из регистров, включение его в пакет DELAY_RESPONSE и передача (мастер)	ПО
Когда пакет DELAY_RESPONSE получен, расчет смещения по времени от текущего мастера, используя значения T1, T2, T3 и T4 (Slave)	ПО

27.5.7.3 Системный таймер и настройки

Системный таймер в модуле 1588 представляет собой 64-разрядный счетчик с прямым счетом, который используется для поддержания значения системного времени, которое состоит из двух отдельных счетчиков по 32 бита каждый. Нижние 32 бита системного таймера называются нижним счетчиком системного таймера, а старшие 32 бита системного таймера называются верхним системным таймером. Если устройство является ведущим в сети 1588, счетчик должен быть инициализирован только один раз в начале. Если устройство является ведомым в сети 1588, счетчик необходимо часто подстраивать после каждого события SYNC и после вычисления значения смещения, как определено ранее, при условии, что текущее значение системного времени отличается от значения времени, сообщенного мастером. Регулировка счетчика может быть выполнена с использованием регистров настройки системного времени, как описано ниже.

Этот счетчик инициализируется при включении питания или при выполнении программного сброса в регистрах DMA, при включении любого из таймстемпингов в регистре Module_1588_Control_Register, после чего счетчик увеличивается на каждый

период приращения, определенный в регистре атрибутов приращения. Значение, которое добавляется к системному таймеру для каждой операции приращения, определяется в значении приращения в регистре атрибутов приращения. Например, при заданной тактовой частоте 1588 (clk_1588) 100 МГц (10 нс):

- Если значение приращения равно 10, а период приращения равен 1, значение 10 добавляется к счетчику каждый такт. Это позволяет счетчику работать с наносекундным разрешением и считать время в наносекундах.

- Если значение приращения равно 100, а период приращения равен 1, значение 100 добавляется к счетчику каждый такт. Это позволяет счетчику, работать с разрешением 0,1 нс, и отсчитывает время с точностью до 0,1 нс.

В зависимости от требований приложения счетчик может быть реализован с любым разрешением, выбрав соответствующие значения для полей “Increment Period” и “Increment Value” в регистре “Increment Attribute Register”.

Для считывания системного таймера ПО должно использовать регистры “System Time Value Lower/Upper”. ПО должно сначала прочитать нижний регистр, после чего старшие 32 бита счетчика фиксируются в верхнем регистре значения системного времени. Это обеспечивает точный способ считывания счетчика при двух отдельных запросах без какой-либо промежуточной операции приращения, влияющей на значение, возвращаемое в ПО.

ПО может инициализировать системный таймер, записав его в “System Time Value Upper Register” и выполнить операцию регулировки для изменения младших 32-разрядных значений системного таймера. Регулировки могут быть выполнены с помощью регистров “System Time Adjust Control Registers”. Нижние 32 бита нижнего регистра “System Time Adjust Control Lower Register” содержат значение, подлежащее настройке (добавляемое или вычитаемое из текущего значения системного таймера), которое следует запрограммировать перед началом процесса настройки. Направление подстройки определяется 31-м битом регистра “System Time Adjust Control Upper Register”. Когда бит [31] равен нулю, выполняется положительная настройка (значение в “System Time Adjust Control Lower Register” добавляется в системный таймер). Когда бит [31] равен единице, выполняется отрицательная настройка (значение в “System Time Adjust Control Lower Register” вычитается из системного таймера).

27.5.7.4 Передача метки времени

Бит “Timestamp Packet Indication” в дескрипторе передачи (TDESC0[30]) используется для указания контроллеру на выполнение отметки времени в текущем пакете. GEMAC будет выполнять отметку времени (извлекается значение системного таймера), когда передается по интерфейсу RGMII первый байт поля DA или первый полубайт поля DA (на скоростях 10/100 Мбит/с). Как только будет выполнена передачи отметки времени, зафиксированное значение системного времени загружается в регистры “Transmit Timestamp” и

устанавливается прерывание, указывающее на успешное завершение операции передачи отметки времени.

После того как пакет помечен меткой времени, никакие операции метки времени не будут выполняться (даже если бит TDESC0[30] установлен для следующих пакетов) до тех пор, пока ПО не прочитает регистры “Transmit Timestamp Registers”. Это позволяет обеспечить взаимосвязь между ПО и контроллером для возврата значения метки времени. Чтобы прочитать значение метки времени, ПО должно прочитать “Transmit Timestamp Lower Register”, а затем “Transmit Timestamp Upper Registers”.

Отметка времени в тракте передачи может быть отключена путем сброса бита разрешения передачи отметки времени в регистре “1588 Control Register”.

27.5.7.5 Прием метки времени

На тракте приема (если включена функция приема отметок времени) контроллер анализирует входящие RX пакеты и проверяет доступные PTP пакеты. Ядро GEMAC может обнаруживать пакеты PTP V1 (пакет PTP в полезной нагрузке UDP через IPV4) или пакеты PTP V2 (пакет PTP в качестве полезной нагрузки L2 или пакет PTP в полезной нагрузке UDP через IPV4 или IPV6). Пакет PTP идентифицируется путем сопоставления поля L2 или UDP Port Number входящего кадра. Как только пакет PTP идентифицирован и совпали MessageID пакета PTP (для PTP V2) или поле управления (для PTP V1), выполняется отметка времени.

В Таблица 27.105 представлен формат сообщения PTP V1 и V2, а поля, в которых заинтересовано ядро GEMAC, выделены жирным шрифтом и подчеркнуты.

Таблица 27.105. Формат сообщения PTP V1 и V2

Byte Offset	V1 PTP Packet	V2 PTP Packet
00	***versionPTP***	{transportSpecific, **messageID**}
01		{reserved, **versionPTP**}
02	*versionNetwork*	messageLength
03		
04	*Subdomain*	domainNumber
05		reserved
06		flags
07		
08		correctionField
09		
10		
11		
12		
13		
14		correctionSubField
15		
16		reserved
17		
18		

Byte Offset	V1 PTP Packet	V2 PTP Packet
19		
20	***messageType***	**Reserved**
21	***Src Comm Tech***	**Src Comm Tech**
22	***sourceID***	**sourceId**
23		
24		
25		
26		
27		
28	***sourceportid***	**sourceportid**
29		
30	***sequenceID***	**sequenceID**
31		
32	***controlField***	controlField
33	*Reserved*	logMessageInterval
34	*flags*	NA
35		

Контроллер использует два бита в дескрипторе приема (RDESC1[31] и RDESC1[30]) для предоставления информации о PTP пакетах и указателях меток времени RX пакета. “PTP Packet Indication” (RDESC1[31]) устанавливается для каждого кадра, который идентифицирован как пакет PTP, в то время как бит “Receive Packet Timestamped” устанавливается только тогда, когда получена метка времени.

Помимо метки времени в програмно доступных регистрах сохраняются SeqID (смещение байтов 30 и 31) и SrcID (смещение байтов 20-29) PTP пакета, и устанавливается прерывание. После приема пакета с меткой времени контроллер продолжит отслеживать и идентифицировать PTP пакеты, но не будет выполнять операции с меткой времени, пока ПО не прочитает регистры “Receive Timestamp Registers”. Это позволяет обеспечить взаимосвязь между ПО и контроллером для возврата значения метки времени. Чтобы прочитать значение метки времени, ПО должно прочитать “Receive Timestamp Lower Register”, а затем “Receive Timestamp Upper Registers”.

27.5.8 IEEE 1588/PTP (одноступенчатый режим)

Контроллер дополнительно поддерживает одношаговый таймстемпинг в дополнение к стандартной двухэтапной процедуре, описанной в предыдущем разделе. Выбор режима работы передатчика определяется программно. Для приемника нет разницы между 1-ступенчатым и 2-ступенчатым режимом.

В 1-ступенчатом режиме нет необходимости в передаче пакета FOLLOW_UP от Clock Master. Пакет SYNC будет содержать информацию о метке времени.

27.5.8.1 Распределение обязанностей контроллера и ПО

В 1-ступенчатом режиме синхронизации времени контроллер GEMAC выполняет следующие функции:

- Определяет (из дескрипторов) TX пакеты требующих метки времени.
- Вставляет точную метку времени в “SYNC PTP Messages” или обновляет поле коррекции.
    - Требуется поле «Смещение» из ПО (через дескриптор передачи)
    - Если пакеты UDP, то обновляется поле “Checksum”. Смещение для поля “Checksum” также указывается ПО.
- Анализирует входящие пакеты и идентифицирует PTP пакеты.
    - Поддерживает форматы кадров IEEE 1588 V1 и V2.
    - V1 PTP в полезной нагрузке UDP через IPV4
    - V2 PTP как специальные Ethertype или UDP через IPV4 или IPV6.
- Идентифицирует RX пакеты, требующие метки времени (на основе идентификатора сообщения (Message ID) PTP).
- Пропускает PTP-пакет и указатели с метками времени для ПО.
- Сохраняет значение принятых меток времени и SeqID/SrcID для ПО.
- Реализует системный таймер с программируемым значением приращения и периодом приращения.
- Предоставляет программный интерфейс для настройки системного таймера.

Ожидается, что ПО будет отвечать за следующие функции:

- Выполнение протокола BMC, что означает определение состояния узла (ведущего или подчиненного) и выбор главных часов, если они находятся в подчиненном состоянии.
- Генерация и анализ PTP-пакетов.
- Расчет смещения времени и непрерывная настройка системного времени с использованием программного интерфейса.

Таблица 27.106 отображает хронологические события потока синхронизации сверху, а также действия, выполняемые ядром GEMAC, и действия, которые должны выполняться программным обеспечением.

Таблица 27.106. Хронологические события потока синхронизации (одоступенчатяй режим)

Действие	Обязанность
Генерация пакета SYNC с индикацией пакета метки времени (master)	ПО
Отметка времени пакета SYNC и предоставление значения отметки времени, T1 (Master)	Контроллер
Отметка времени входящего пакета SYNC, T2 и предоставление его вместе с деталями пакета PTP (SeqID / SrcID) и указателями отметки времени (slave)	Контроллер
Генерация пакета DELAY_REQ с указанием пакета метки времени в дескрипторе передачи (slave)	ПО
Отметка времени пакета DELAY_REQ и предоставление значения отметки времени, T3 (Slave)	Контроллер
Отметка времени входящего пакета DELAY_REQ, T4 и предоставление его вместе с деталями пакета PTP (SeqID/SrcID) и индикацией отметки времени (Master)	Контроллер
Чтение значения принятой метки времени из регистров, включение его в пакет DELAY_RESPONSE и передача (мастер)	ПО
Когда пакет DELAY_RESPONSE получен, расчет смещения по времени от текущего мастера, используя значения T1, T2, T3 и T4 (Slave)	ПО

27.5.9 Счетчики статистики

Ядро имеет несколько 32-разрядных счетчиков для поддержки RMON/SNMP/802.3 MIB статистики. Для оптимизации площади счетчики хранятся в памяти внутри ядра. Управляющая логика выполняет операции чтения-модификации-записи счетчиков для каждого переданного и принятого пакета. Счетчики разделяются на счетчики приема и счетчики передачи. Также имеется два счетчика, хранящие информацию о Receive DMA.

Счетчики можно читать, используя регистры Transmit/Receive StatCtr Data/Control Registers. StatCtr Control Registers отображает номер читаемого регистра. Далее описаны функции различных счетчиков.

27.5.9.1 Счетчики DMA (DMA Counters)

Соответствующие счетчики обновляются, если DMA не может передать в память принятые пакеты по причине отсутствия дескриптора приема или по причине нахождения DMA в состоянии STOPPED. Читая данный счетчик, можно определить количество пропущенных кадров по причине программных ошибок. Эти счетчики сбрасываются при чтении и формируют запрос на прерывание при переполнении.

Missed Frame Counter: Этот счетчик инкрементируется каждый раз, когда новый принятый по Ethernet интерфейсу кадр не может быть перенесен DMA в память из-за отсутствия дескриптора приема.

Stop Flush Counter: Этот счетчик инкрементируется каждый раз, когда новый принятый по Ethernet интерфейсу кадр не может быть перенесен DMA в память из-за того, что DMA находится в состоянии STOPPED.

27.5.9.2 Счетчики передатчика (Transmit Counters)

Счетчики передатчика используются для сбора статистики по операциям передачи на MII/GMII интерфейсе. Эти счетчики отображают SNMP/RMON статистику. Все счетчики передачи размещены в памяти ядра и обновляются каждый раз после выполненной передачи кадра. Эти счетчики 32-разрядные, при переполнении запрос на прерывание не формируется, а просто начинается счет заново. Сброс счетчиков осуществляется только при выключении питания.

В Таблица 27.107 указаны счетчики передачи с их смещением, которое заносится в регистр StatCtr Control Register.

Таблица 27.107. Перечень статистических счетчиков передачи

Смещение Tx. Counter	Имя Tx. Counter	Описание Transmit Counter
00	FramesTransmittedOk	Счетчик количества кадров, переданных без ошибок, включая пакеты PAUSE.
01	FramesTransmittedTotal	Счетчик общего количества переданных пакетов.
02	OctetsTransmittedOk	Отображает общее количество байт в кадрах, переданных без ошибок. Учитываются все поля кроме PREAMBLE и SFD.
03	FramesTransmittedErrpr	Счетчик количества кадров, переданных с ошибкой (EOP/ERR).
04	FramesTransmittedSingleClsn	Счетчик количества успешно переданных кадров после единичного столкновения. Счетчик достоверен только в режиме Half-Duplex.
05	FramesTransmittedMultipleClsn	Счетчик количества успешно переданных кадров после множественного столкновения. Счетчик достоверен только в режиме Half-Duplex.
06	FramesTransmittedLateClsn	Счетчик количества пакетов, на которых произошла поздняя коллизия. Пакет может быть оборван. Счетчик достоверен только в режиме Half-Duplex.
07	FramesTransmittedExcessiveClsn	Счетчик количества пакетов, на которых произошло чрезмерное количество коллизий. Пакет может быть оборван. Счетчик достоверен только в режиме Half-Duplex.
08	FramesTransmittedUnicastAddr	Счетчик количества пакетов с индивидуальным адресом.
09	FramesTransmittedMulticastAddr	Счетчик количества пакетов с групповым адресом.
10	FramesTransmittedBroadcastAddr	Счетчик количества пакетов с широковещательным адресом.
11	FramesTransmittedPAUSE	Счетчик количества переданных пакетов PAUSE.

27.5.9.3 Счетчики приемника (Receive Counters)

Счетчики передатчика используются для сбора статистики по операциям приема на MII/GMII интерфейсе. Эти счетчики отображают SNMP/RMON статистику. Все счетчики приема размещены в памяти ядра и обновляются каждый раз после выполненного приема кадра. Эти счетчики 32-разрядные, при переполнении запрос на прерывание не формируется, а просто начинается счет заново. Сброс счетчиков осуществляется только при выключении питания.

В Таблица 27.108 указаны счетчики приема с их смещением, которое заносится в регистр StatCtr Control Register.

Таблица 27.108. Перечень статистических счетчиков приема

Смещение Tx. Counter	Имя Tx. Counter	Описание Transmit Counter
00	FramesReceivedOK	Счетчик безошибочно принятых кадров
01	FramesReceivedTotal	Общее количество принятых пакетов.
02	FramesReceivedCRCErr	Общее количество принятых пакетов с ошибкой CRC и других пакетов типа Ethernet.
03	FramesReceivedAlignErr	Счетчик количества принятых пакетов с Alignment Error. Это возможно только при работе с MII интерфейсом.
04	FramesReceivedError	Счетчик пакетов принятых с ошибками: CRC Error Alignment Error MaxFrameLen Error Receive Error Length Error MinFrameSize Error
05	OctetReceivedOK	Счетчик количества байт в пакетах, принятых без ошибок. Для каждого пакета подсчитываются все байты, кроме PREAMBLE и SFD.
06	OctetReceivedTotal	Счетчик количества байт во всех принятых пакетах.
07	FramesReceivedUnicastAddr	Счетчик количества пакетов, принятых без ошибок, с индивидуальным адресом.
08	FramesReceivedMulticastAddr	Счетчик количества пакетов, принятых без ошибок, с групповым адресом.
09	FramesReceivedBroadcastAddr	Счетчик количества пакетов, принятых без ошибок, с широковещательным адресом.
10	FramesReceivedPAUSE	Счетчик количества принятых пакетов PAUSE.
11	FramesReceivedLengthErr	Счетчик количества принятых пакетов с ошибкой длины.
12	FramesReceivedUnderSized	Счетчик количества принятых пакетов с ошибкой minFrameSize (длина меньше 64 байт), но с верным значением FCS.
13	FramesReceivedOverSized	Счетчик количества принятых пакетов с ошибкой maxFrameSize (значение, запрограммированное в регистре Max Frame Size Register), но с верным значением FCS.
14	FramesReceivedFragments	Счетчик количества принятых пакетов с ошибкой minFrameSize (длина меньше 64 байт) и ошибкой в FCS. Возникает, когда принят фрагмент кадра, образовавшийся при коллизии.
15	FramesReceivedJabber	Счетчик количества принятых пакетов с ошибкой maxFrameSize и с ошибкой в FCS. Этот счетчик не имеет ничего общего с регистром Receive Jabber Size Register использующегося для обрезания слишком больших пакетов.
16	FramesReceivedLen64	Счетчик количества принятых пакетов с длиной 64 байта.
17	FramesReceivedLen65-127	Счетчик количества принятых пакетов с длиной от 65 до 127 байт.
18	FramesReceivedLen128-255	Счетчик количества принятых пакетов с длиной от 128 до 255 байт.
19	FramesReceivedLen256-511	Счетчик количества принятых пакетов с длиной от 256 до 511 байт.
20	FramesReceivedLen512-1023	Счетчик количества принятых пакетов с длиной от 512 до 1023 байт.
21	FramesReceivedLen1024-1518	Счетчик количества принятых пакетов с длиной от 1024 до 1518 байт.

Смещение Tx. Counter	Имя Tx. Counter	Описание Transmit Counter
22	FramesReceivedLen1519Plus	Счетчик количества принятых пакетов с длиной от 1518 байт до значения, запрограммированного в регистре Max Frame Length Reister.
23	FramesDroppedBufferFull	Счетчик количества пакетов сброшенных из-за переполнения Receive FIFO. Это случается, когда DMA не может обеспечить достаточную скорость выгрузки данных.
24	FramesTruncatedBufferFull	Счетчик количества пакетов, усеченных из-за переполнения Receive FIFO при работе в Cut-Thru режиме. Эта ситуация возникает, когда DMA начал передачу принимаемого пакета, но при этом весь пакет не поместился в FIFO.

27.5.10 Поддержка VLAN

GEMAC ядро поддерживает определение тега VLAN в принятом VLAN пакете. Если в принятом пакете определен тег VLAN, то в поле статуса пакета последнего дескриптора отмечается бит VLAN Frame. GEMAC ядро анализирует входящий кадр и может обнаружить до 3 тегов VLAN (Q-in-Q) прежде, чем будет извлечено поле Length/Type. GEMAC ядро может быть запрограммировано на автоматическую настройку поля MaxFrameLen для учета 4-байтового поля VLAN. При получении VLAN пакета поле MaxFrameLen увеличивается на 4 индивидуально для каждого VLAN тега.

27.5.11 Обработка ошибок

В следующих разделах описаны варианты обработки ошибок, возникающих в GEMAC ядре. Эти ошибки не определены в спецификации IEEE 802.3 и возникают либо при ошибке программирования, либо при возникновении нештатных ситуаций в контроллере, при этом данные ошибки можно исправить. В зависимости от направления передачи ошибки классифицируются на ошибки приема и ошибки передачи.

27.5.11.1 Transmit EOP/Error

Если в дескрипторе передачи установлен бит ForceEOPErr (TDES1[24]), то кадр отправляется с EOP/Error. Передающая логика MAC генерирует пакет, в конец которого вставляется инвертированный CRC. В случае если пакет выгружается из памяти уже с полем FCS, то контроллер добавляет еще одно поле с инверсией для отображения ошибки. Это программный путь формирования ошибки в кадре. При этом счетчик статистики FramesTransmittedError инкрементируется при каждой генерации пакета с EOP/ERR.

27.5.11.2 Ошибка недогрузки (Transmit Underrun Error)

Если во время передачи кадра Transmit FIFO оказывается пустым, то возникает состояние недогрузки. Причиной этому может быть, как недостаточная скорость, с которой DMA

загружает данные в Transmit FIFO, так и отсутствие доступного дескриптора при описании одного пакета несколькими дескрипторами.

Если такое состояние возникло, то передатчик обрывает передачу пакета и формирует признак ошибки EOP/ERR (в последнем такте устанавливается TX_ER). Контроллер в таком кадре не добавляет поле FCS. При возникновении подобного события счетчик статистики передачи FramesTransmittedError инкрементируется, формируется запрос на прерывание. Когда непереданные данные догрузятся в Transmit FIFO, контроллер их сбросит.

27.5.11.3 Ошибка бессмысленной передачи (Transmit Jabber Error)

Для определения бессмысленный передается кадр или нет, служит Transmit Jabber Register. Кадр считается бессмысленным, если его длина превышает значение, запрограммированное в регистре Transmit Jabber Register. Это случается при некорректном формировании дескриптора передачи. GEMAC ядро обрывает передачу пакета сразу, как только обнаруживается, что данные бессмысленны, и устанавливает EOP/ERROR (в последнем такте устанавливается TX_ER). Контроллер в таком кадре не добавляет поле FCS. При возникновении подобного события счетчик статистики передачи FramesTransmittedError инкрементируется, формируется запрос на прерывание. Оставшиеся от пакета данные сбрасываются из Transmit FIFO.

27.5.11.4 Чрезмерное количество коллизий передачи (Transmit Excessive Collision)

Если при работе в режиме Half-Duplex контроллер обнаруживает коллизию на 16-й повторной передачи, то передача кадра прерывается, кадр сбрасывается из FIFO и формируется признак ошибки чрезмерного количества столкновений. Счетчик столкновений очищается, контроллер переходит в ожидание следующей передачи, и инкрементируется статистический счетчик FrameTransmittedExcessiveClsn.

27.5.11.5 Поздняя коллизия передачи (Transmit Late Collision)

Если контроллер, работая в режиме Half-Duplex, обнаруживает позднюю коллизию (столкновение произошло после передачи 64 байт), то передача пакета прекращается. После обработки столкновения (передача JAM сообщения) фрейм сбрасывается из Transmit FIFO и повторная передача не производится. Счетчик столкновений очищается, контроллер переходит в ожидание следующей передачи, и инкрементируется статистический счетчик FrameTransmittedLateClsn.

27.5.11.6 Ошибка приема PREAMBLE/SFD (Receive PREAMBLE/SFD Errors)

Когда приемник MAC обнаруживает ошибку в преамбуле или SFD байте, прием пакета останавливается до следующего кадра. Ни один из счетчиков статистики приема при возникновении данной ошибки не обновляется.

27.5.11.7 Прием кадра с ошибкой (Receive Frame Errors)

При обнаружении любой ошибки во входящем кадре приемник сообщает об ошибке Receive FIFO и ошибке Receive DMA. В режиме Store-and-Forward пакет сбрасывается из Receive FIFO без трансляции в память. Но если установлен бит PassBadFrames, то независимо от режима работы (Store-and-Forward или Cut-Thru) пакет с ошибкой передается в память с пометкой соответствующего ошибке бита статуса DMA (RDES0). Также контроллер обновляет соответствующий типу ошибки счетчик статистики.

27.5.11.8 Прием короткого пакета (Receive Runt Frame)

Если длина принятого пакета меньше 64 байт, то пакет объявляется коротким и сбрасывается из Receive FIFO при работе в режиме Store-and-Forward. При работе в режиме Cut-Thru кадр сбрасывается, если значение Rx Packet Start Threshold меньше или равен 64 байтам. Счетчик, соответствующий ошибке приема, обновляется только в том случае, если прошла проверка CRC.

27.5.11.9 Прием бессмысленного пакета (Receive Jabber Frame)

Для определения бессмысленный текущий кадр или нет, в ядре используется Receive Jabber Register. Если длина принимаемого пакета превышает значение, запрограммированное в регистре Receive Jabber Register, то кадр признается бессмысленным.

Если ядро детектировало прием бессмысленного кадра, то прием останавливается до следующего кадра. Часть кадра, которая была принята, сбрасывается из FIFO либо передается в память с пометкой Jabber Error.

27.5.11.10 Переполнение Receive FIFO (Receive FIFO Full)

Если Receive FIFO переполнился, то прием прекращается и пакет сбрасывается из FIFO. Переполнение может возникнуть при недостаточной скорости выгрузки данных в память. При сбросе пакета из-за переполнения FIFO инкрементируется счетчик FrameDroppedBufferFull. В режиме Cut-Thru начальная часть пакета, принятая до переполнения, может сохраниться, оставшийся пакет обрывается. В этом случае инкрементируется счетчик FrameTruncatedBufferFull.

27.5.12 Запросы на прерывание

При возникновении ряда событий во время приема и передачи пакетов GEMAC формирует запрос на прерывание. Узнать причину запроса на прерывание можно, прочитав регистры DMA Status и IRQ Register. Все биты запросов на прерывание сбрасываются записью в них единицы. Запрос на прерывание транслируется на AHB интерфейс, если установился какой-либо бит IRQ, и установлено соответствующее ему разрешение. Снять запрос на прерывание с AHB интерфейса можно только очисткой всех битов, вызывающих запрос на прерывание. В контроллере возможно возникновение прерываний по следующим событиям.

27.5.12.1 Передача выполнена

Бит Transmit Transfer Done IRQ устанавливается в том случае, если пакет, прочитанный из памяти, успешно записан в Transmit FIFO (установлен бит Wait for Done в DMA Configuration Register) и успешно передан на MII интерфейс. Запрос на прерывание формируется только в том случае, если в первом дескрипторе пакета установлен бит Interrupt on Completion (TDES[31]).

27.5.12.2 Дескриптор передачи недоступен

Бит Transmit Descriptor Unavailable IRQ устанавливается в том случае, если следующий по списку дескриптор передачи не готов для использования. Transmit DMA приостанавливается. Для того чтобы возобновить выгрузку дескрипторов передачи, необходимо осуществить запись в регистр Transmit Poll Demand Register, если не включен Transmit Auto Polling.

27.5.12.3 Transmit DMA остановлено

Если Transmit DMA находится в состоянии STOPPED, то устанавливается бит Transmit DMA Stopped. Если во время работы DMA приходит программная команда перехода в состояние STOP, то она откладывается до окончания передачи кадра и закрытия дескриптора.

27.5.12.4 Прием выполнен

Бит Receive Transfer Done IRQ устанавливается в том случае, если пакет принят по MII интерфейсу, успешно перемещен в память и последний дескриптор закрыт с указанием статуса принятого кадра. По умолчанию Receive Transfer Done IRQ устанавливается для каждого пакета, переданного в память. Запрос на прерывание устанавливается только после закрытия последнего дескриптора.

Receive Interrupt Mitigation Register используется для минимизации количества запросов на прерывание. Если функция уменьшения количества запросов на прерывание включена, то

запрос формируется не после каждого принятого и переданного в память пакета, а после определенного количества таких кадров, указанного в поле Frame Count.

Если кадры транслируются с большим интервалом, то в работу включается таймер (Time-Out counter), отсчитывающий время с момента приема первого кадра до приема количества кадров, указанного в Frame Count. Если таймаут наступает быстрее, то формируется запрос на прерывание.

Существует возможность сброса Frame Count в 1 после установки запроса на прерывание, если запрограммировано очень большое значение Frame Count.

27.5.12.5 Дескриптор приема недоступен

Бит Receive Descriptor Unavailable IRQ устанавливается в том случае, если следующий по списку дескриптор приема не готов для использования. Receive DMA приостанавливается. Для того чтобы возобновить выгрузку дескрипторов приема, необходимо осуществить запись в регистр Receive Poll Demand Register. Если нет записи в регистр Receive Poll Demand Register, то DMA приема обратится за новым дескрипторов сам при обнаружении нового принимаемого кадра.

27.5.12.6 Receive DMA остановлено

Если Receive DMA находится в состоянии STOPPED, то устанавливается бит Receive DMA Stopped. Если во время работы DMA приходит программная команда перехода в состояние STOP, то она откладывается до окончания приема кадра, его передачи в память и закрытия дескриптора.

27.5.12.7 Пропуск кадра

Receive Missed Frame Interrupt формируется, если кадр сбрасывается по причине отсутствия дескриптора приема. Фрейм сбрасывается из Receive FIFO. Данный бит устанавливается только в том случае, когда Receive DMA находится в состоянии SUSPENDED и принят новый пакет, который провоцирует DMA выбрать дескриптор, а дескриптор все еще не готов.

27.5.12.8 Недозагрузка во время передачи

Underrun Interrupt выставляется при нехватке данных, передаваемых на MII интерфейс. Это возникает, когда DMA не может поддержать скорость передачи данных, требуемую MAC. Контроллер прекращает передачу, выставляет на MII сигнал EOP/ERR, а оставшиеся данные игнорирует.

27.5.12.9 Передача бессмысленных данных

Jabber Interrupt выставляется при обнаружении на передатчике бессмысленных данных (длина пакета больше значения запрограммированного в регистре Transmit Jabber Count Register). Контроллер прекращает передачу, выставляет на MII/GMII сигнал EOP/ERR, а оставшиеся данные игнорирует.

27.5.13 Удаленное пробуждение

В контроллере реализована функция удаленного пробуждения через прием “Magic Packet” или направленного кадра с уникальным адресом. Направленный кадр с уникальным адресом – это любой кадр, поле DA которого соответствует регистру “MAC Address#0 Register”. Пакет “Magic Packet” основан на технологии AMD для пробуждения спящего сетевого устройства.

Когда в контроллере включен режим пробуждения, ядро GEMAC постоянно отслеживает кадры пробуждения. Полученные кадры не передаются в FIFO, а сбрасываются, поэтому системные частота и частота передатчика.

После получения кадра пробуждения контроллер генерирует событие пробуждения (WkupRcvd). WkupRcvd сбрасывается после перевода контроллера на работу в обычном режиме (бит “Wakeup Mode” сброшен). Если не сбросить WkupRcvd контроллер будет продолжать отбрасывать принятые кадры, не записывая их в принимающее FIFO, а также остановится мониторинг кадров пробуждения.

27.5.13.1 Пробуждение направленным кадром с уникальным адресом

Этот режим включается установкой бита “Directed Unicast Wakeup Mode” в регистре “MAC Control Register”. Когда этот режим включен, GEMAC блокирует запись всех кадров в принимающее FIFO и отслеживает принятый без ошибок направленный кадр с уникальным адресом. Направленный кадр с уникальным адресом – это Uncast пакет поле DA которого совпадает с содержимым регистра “MAC Address#0”. Как только направленный кадр с уникальным адресом получен, подается сигнал WkupRcvd, указывающий на прием кадра пробуждения. Этот режим можно отключить путем сброса бита “Directed Unicast Mode”.

27.5.13.2 Пробуждение кадром «Magic Packet»

Этот режим включается установкой бита “Magic Packet Wakeup Mode” в регистре “MAC Control Register”. Когда этот режим включен, GEMAC блокирует запись всех кадров в принимающее FIFO и отслеживает принятый без ошибок “Magic Packet”, адресованный этому устройству. Только те кадры, которые проходят фильтрацию адресов (соответствует одному из запрограммированных регистров MAC-адресов) или широковещательный кадр, обрабатываются для обнаружения “Magic Packet”. “Magic Packet” – это пакет с шаблоном синхронизации 48’hFF_FF_FF_FF_FF_FF в полезной нагрузке кадра, за которым сразу

следуют 16 повторений MAC-адреса устройства без каких-либо разрывов или прерываний. Шаблон синхронизации может начинаться в любом месте полезной нагрузки кадра, но за ним должны немедленно следовать 16 повторений MAC-адреса. Контроллер использует значение, запрограммированное в регистре “MAC Address#0”, для сравнения 16 повторений адреса. Как только “Magic Packet” получен, формируется сигнал WkupRcvd, указывающий на прием кадра пробуждения.

Этот режим можно отключить, сбросив бит “Magic Packet Wakeup Mode”.

28. КОНТРОЛЛЕР SD/EMMC (SDMMC)

28.1 Основные характеристики модуля SD/MMC

System/Host интерфейс:

- передача данных осуществляется в двух режимах: PIO-режим на Slave-интерфейсе шины, DMA-режим — на Master-интерфейсе шины.

SD/ SDIO Card интерфейс:

- частота клока варьируется от 0 до 208 МГц;
- при скорости до 832 Мбит/сек используется параллельный четырехпроводной интерфейс (режим SDR104);
- передача данных в одно- и четырехбитном SD режиме и в SPI режиме;
- поддерживаются SDR104, SDR50, DDR50 режимы;
- алгоритм проверки контрольной суммы CRC7 для проверки целостности команд и CRC16 для проверки целостности данных;
- передачи данных переменной длины;
- поддерживает управление ожиданием чтения, приостановка/возобновление работы SD/SDIO;
- поддерживается работа с I/O картами, Read-only картами и Read/Write картами.

MMC Card интерфейс:

- частота клока варьируется от 0 до 208 МГц;
- при скорости до 1664 Мбит/сек используется параллельный восьми-проводной интерфейс (mmc 8 битный SDR режим);
- при скорости до 832 Мбит/сек используется параллельный восьми-проводной интерфейс (mmc 8 битный DDR режим);
- передача данных в одно- четырех- и восьми-битном режиме и SPI режиме;
- алгоритм контроля контрольной суммы CRC7 для проверки целостности команд, и CRC16 для проверки целостности данных;
- поддержка MMC Plus и MMC Mobile;
- обнаружение карты (установлена/извлечена);
- максимальная адресация – 2 Тбайта.

Данный блок соответствует следующим версиям спецификаций:

- SD Host Controller Standard Specification Version 3.00
- SDIO card specification Version 3.0
- SD Memory Card Specification Version 3.01

- SD Memory Card Security Specification version 1.01
- MMC Specification version 4.51

Контроллер поддерживает карты SD до 2 Гбайт, SDHC от 2 Гбайт до 32 Гбайт, SDXC от 32 Гбайт до 2 Тбайт.

Карты SDUC не поддерживаются (контролер работает по стандарту MMC Specification version 4.51).

28.1.1 Хост интерфейс (Master/Target(Slave))

Шина Target используется для получения доступа к внутренним регистрам контроллера в режиме PIO (Programmed Input/Output).

Шина Master используется DMA контроллером (в DMA и ADMA2 режимах) для передачи данных между внутренними регистрами SD контроллера и системной памятью.

28.2 Описание модулей, входящих в ядро SDHC

Структурная схема SD хост контроллера представлена на рисунке 20.2. В состав SDHC входят следующие модули:

28.2.1 Host Controller Register Set (набор регистров хост контроллера)

Набор внутренних регистров SD контроллера. Обеспечивает организацию управляющих сигналов для остальных блоков устройства и контролирует состояние сигналов от блоков, устанавливая биты состояний прерываний. Генерирует сигнал прерывания на шину хоста.

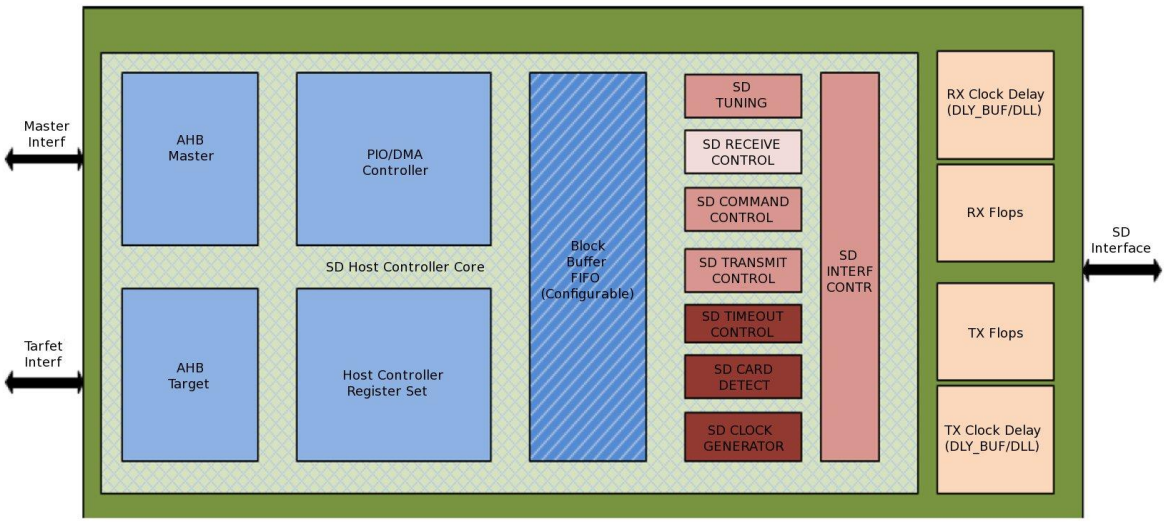


Рисунок 28.1. Структурная схема ядра SDHC

28.2.2 PIO/DMA Controller

Данный модуль реализует функционирование SDMA и ADMA2, а также поддерживает правильную работу модуля в PIO режиме. Он взаимодействует с набором внутренних регистров и организует работу DMA. DMA контролер с одной стороны подключен к системной Master шине, для осуществления передачи данных. С другой стороны он подключен к внутреннему буферу Block Buffer для осуществления операций хранения/извлечения блоков данных.

28.2.3 Block Buffer

Внутренний двух портовый буфер, который используется для хранения блоков данных при осуществлении SD трансфера. Имеет циркулярную буферизацию. С одной стороны буфер имеет соединение с DMA контроллером и тактируется клоком хоста, с другой стороны он соединен с управляющей SD логикой и тактируется SD клоком. Во время осуществления записи на карту, данные извлекаются из системной памяти и помещаются в этот буфер. Когда блок данных становится доступен, управляющая SD логика передает его на SD интерфейс. Если в буфере есть свободное место, DMA контроллер передает следующий блок данных.

Во время операции чтения с карточки, блок данных помещается в буфер и проверяется его контрольная сумма. Затем DMA контроллер передает этот блок в системную память. Тем

временем, при наличии свободного места в буфере, SD логика передает следующий блок данных. Если контролер не может принять данные с карточки, то он подаст сигнал на ожидание чтения (если карточка поддерживает этот механизм) или остановит клок для приостановки передачи данных с карты.

28.2.4 SD Clock Generator

Модуль генерирует частоту CLK интерфейса SD из опорной частоты (xin_clk) в соответствии с настройками регистра Clock Control. В регистре Clock Control задаются коэффициент деления, признак включение частоты и др. настройки.

28.2.5 SD Card Detect

Этот модуль контролирует наличие/отсутствие SD карты, считывая сигнал с пина SD_CD#. В этом модуле осуществляется фильтрация с целью защиты от ложного срабатывания, вызванного дребезгом контактов.

28.2.6 SD Timeout Control

Этот модуль реализует проверку таймаутов во время пересылки данных. Работает под контролем модулей передачи и приема данных.

28.2.7 SD Command Control

Этот модуль генерирует последовательность команд на CMD линию SD интерфейса. Данный модуль также осуществляет прием ответа, проводит проверку его правильности (наличие Start-бита, номера команды, CRC-суммы, End-бита) и время прихода (таймаут). Далее результаты ответа передаются в модуль внутренних регистров для установки различных статусных битов. Также здесь генерируются команды управления для модулей передачи и приема данных.

28.2.8 SD Transmit Control

Модуль осуществляет операцию записи передаваемых на SD карту данных. После получения команды, модуль ждет, когда блок данных будет доступен в буфере, после чего передает его на DAT линию SD интерфейса. В зависимости от конфигурации линии передачи данных (1-, 4-, 8-битная или SPI) данные из буфера соответствующим образом распределяются. Рассчитывается CRC16 и помещается в конце блока, перед end-битом. В случае DDR операций контрольная сумма рассчитывается для каждого уровня клока. В конце передачи блока осуществляется ожидание ответа на линии DAT0, и результат

проверки ответа передается в набор регистров. Также модуль осуществляет проверку Write Busy перед началом передачи нового блока данных.

28.2.9 SD Receive Control

Модуль осуществляет операцию чтения получаемых с SD карты данных. После приема с SD карты блока данных перед их записью в буфер, модуль осуществляет их сборку из (1-, 4-, 8-битного представления) в 32-битные слова. Рассчитывается и сверяется с принятой контрольная сумма CRC16. Также реализуется проверка таймаутов для соблюдения определенной величины задержек.

28.2.10 SD Tuning Block

Этот блок используется в режимах SDR104 (или SDR50) и Hs200 для подстройки принимаемого клока. Подстроечный блок генерирует команды задержки для внешнего модуля управления задержками.

28.2.11 SD Interface Control

Модуль осуществляет передачу внутренних сигналов на внешний SD интерфейс и наоборот. В зависимости от ширины шины (1/4/8) внутренние сигналы преобразуются соответствующим образом.

28.2.12 RxClk Delay Module

Данный модуль используется для поддержки принимаемого клока выравненным по центру относительно принимаемых данных. Существует два режима задержки принимаемого тактового сигнала. Первый из них — автоматическая настройка получаемого клока, используется в режиме SDR104 (для SD 3.0 или eMMC 4.51) и для SDR50 (опционально). Второй — режим ручного управления задержками. Ручной режим управления осуществляется для HS режимов: SDR25/SDR50/DDR50. Данный модуль реализован с помощью линии задержки с отводами (Tap Delay Lines, см. Рисунок 28.2, Рисунок 28.3) для создания клоков с 32 различными фазами и выбора подходящего. Максимальная задержка сигнала при всех включенных элементах задержки составляет 3ns. Управление этим функционалом осуществляется при использовании следующих сигналов:

- *corectrl_itapdlysel[4:0]* — используется для выбора оптимального количества отводов, определяющих нужную задержку;
- *corectrl_itapdlyen* — используется для включения режима управления задержками с помощью отводов.

28.2.13 RX Flops Module

Модуль, где данные с линий CMD/DAT SD интерфейса защелкиваются по тактовому сигналу с выхода от модуля RxClk Delay. Модуль RX Flops следует размещать как можно ближе к I/O для достижения наилучших временных характеристик принимаемых сигналов. Для поддержки обоих DDR режимов, сигналы защелкиваются на обоих фронтах принимаемого клока. Далее данные с выхода этого модуля передаются на обработку в основной блок для дальнейшей обработки.

28.2.14 TxClk Delay Module

Сигналы на выходах CMD и DAT должны задерживаться относительно выходного клока SD_CLK в соответствии с требованиями к времени удержания в различных режимах работы. Задержанный здесь исходящий SD клок используется для управления изменения сигналов CMD/DAT и других сигналов SD интерфейса. Непосредственно сам выход SD_CLK не задерживается. Данный модуль реализован с помощью линии задержки с отводами (Tap Delay Lines, см. Рисунок 28.2, Рисунок 28.3) для создания клоков с 16 различными фазами и выбора подходящего. Максимальная задержка сигнала при всех включенных элементах задержки составляет 3ns.Управление осуществляется с помощью следующих сигналов:

- *corectrl_otapdlysel[3:0]* — используется для выбора оптимального количества отводов, определяющих нужную задержку;
- corectrl_otapdlyen — используется для включения режима управления задержками с помощью отводов.

28.2.15 TX Flops Module

Выходные сигналы с основного блока контроллера защелкиваются по задержанному тактовому сигналу с выхода от модуля TxClk Delay. Также для поддержки обоих DDR режимов, принятые из основного блока контроллера данные передаются как по положительному, так и по отрицательному фронтам задержанного клока.

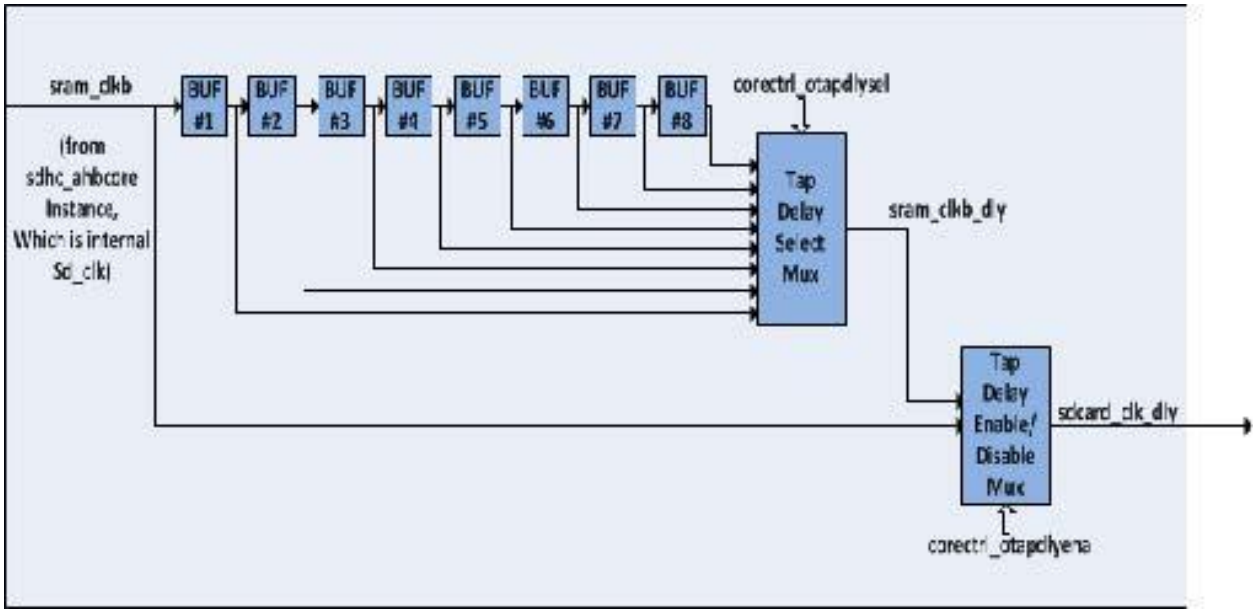


Рисунок 28.2.

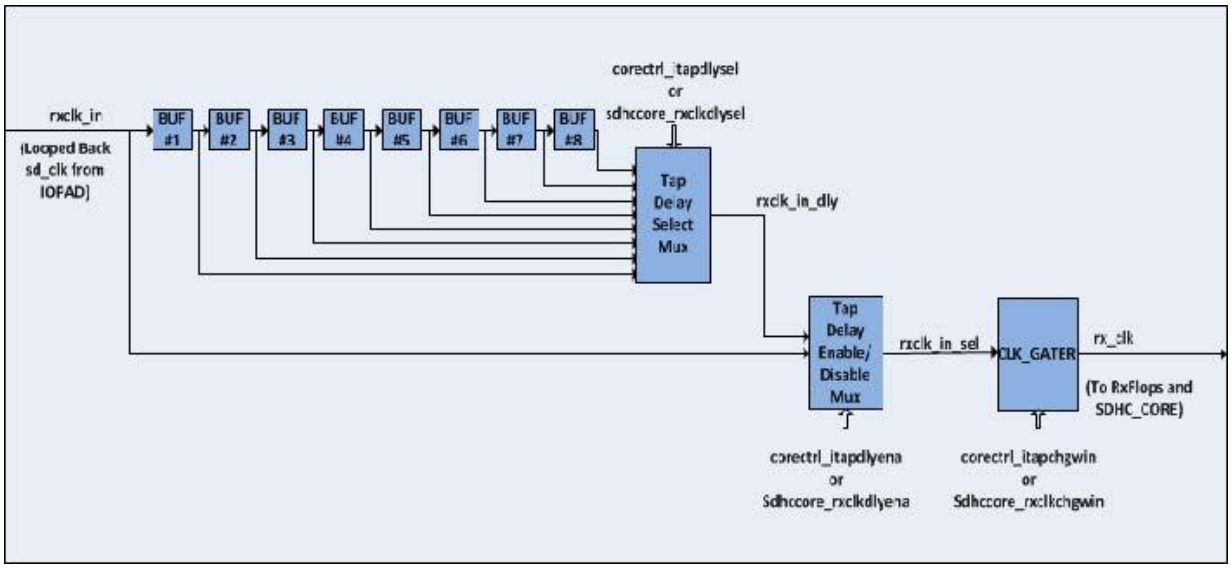


Рисунок 28.3.

28.3 Внутренние регистры ядра SDHC

Таблица 28.1. Карта регистров ядра SDHC

Смещение	Биты 15-08 Биты 07-00	Описание
000h	SDMA System Address / Argument 2 (младшая часть)	Этот регистр содержит адрес системной памяти для SDMA-передач или второй аргумент Auto CMD23
002h	SDMA System Address / Argument 2 (старшая часть)	

Смещение	Биты 15-08	Биты 07-00	Описание
004h	Block Size		Содержит размер блока данных для DMA/CMD
006h	Block Count		Содержит количество блоков при многоблочной передаче
008h	Argument1 (младшая часть)		Аргумент SD команды
00Ah	Argument1 (старшая часть)		
00Ch	Transfer Mode		Регистр используется при передаче данных
00Eh	Command		Регистр используется при передаче команд
010h	Response 0		Регистры используются при ответе на команды
012h	Response 1		...
014h	Response 2		...
016h	Response 3		...
018h	Response 4		...
01Ah	Response 5		...
01Ch	Response 6		...
01Eh	Response 7		...
020h	Buffer Data Port 0		Через эти регистры можно получить доступ к буферу хост контроллера
022h	Buffer Data Port 1		...
024h	Present State		Регистр указывает, какие используются DAT линии SD интерфейса
026h	Present State		...
028h		Host Control 1	Регистр настроек хоста 1
029h	Power Control		Регистр управления питанием
02Ah		Block Gap Control	Регистр управления загрузкой и восстановлением передачи данных
02Bh	Wakeup Control		Регистр управления пробуждением контроллера
02Ch	Clock Control		Регистр используется для выбора частоты SDCLK
02Eh		Timeout Control	Регистр определяет время, по истечении которого на линии DAT детектируется таймаут
02Fh	Software Reset		Регистр управления программным сбросом
030h	Normal Interrupt Status		Регистр отображает состояния стандартных прерываний
032h	Error Interrupt Status		Регистр отображает состояния прерываний, вызванных ошибками
034h	Normal Interrupt Status Enable		Регистр управления разрешениями стандартных прерываний
036h	Error Interrupt Status Enable вызванных ошибками		Регистр управления разрешениями прерываний, вызванных ошибками
038h	Normal Interrupt Signal Enable		Регистр используется для выбора, какие статусы стандартных прерываний будут отображаться в системе
03Ah	Error Interrupt Signal Enable		Регистр используется для выбора, какие статусы прерываний вызванных ошибками будут отображаться в системе
03Ch	Auto CMD12 Error Status		Регистр используется для отображения наличия ошибок в Auto CMD
03Eh	Host Control 2		Регистр настроек хоста 2
040h-047h	Capabilities		Регистр, обеспечивающий реализацию HC
048h-04F	Maximum Current Capabilities		Регистр устанавливает максимальное потребление для различных напряжений
050h	Force Event for Auto CMD Error Status		Регистр принудительно устанавливает прерывания, связанные с появлением ошибок в Auto CMD
052h	Force event for Error Int Status		Регистр принудительно устанавливает статус вызванного ошибкой прерывания

Смещение	Биты 15-08 Биты 07-00	Описание
054h	ADMA Error Status	Регистр содержит статус ошибки при ее появлении в ADMA
056h	–	Резерв
058h-05Fh	ADMA System address [63:0]	Регистр содержит адрес выполняемой команды в дескрипторной таблице
060h-06Fh	Preset Value	Предустановленные значения для карты и контроллера
070h	Boot data timeout control	Регистр определяет время, по истечении которого на линии DAT детектируется таймаут во время загрузки с карты
...		
0E0h-0E2h	Shared bus control	Не реализовано в данной версии контроллера
...		
0FCh	Slot Interrupt Status	Регистр показывает состояние прерывание от слота
0FEh	Host Controller Version	Регистр отображает версию контроллера
100h-11Ch	Конфигурационные сигналы	В данных регистрах устанавливаются значения для аппаратно конфигурируемых регистров, имеющих тип Hwinit

В следующей таблице представлены типы используемых регистров:

Таблица 28.2. Сокращенные обозначения типов регистров

Обозначение типа	Описание
RO	Регистры, доступные только для чтения. Биты регистра не могут быть изменены программно или с помощью операций сброса. Запись в эти биты игнорируется
ROC	Биты этих регистров обнуляются во время сброса. Запись в эти биты игнорируется
RW	Регистры, доступные как на запись, так и на чтение. Биты регистра могут быть изменены программно.
RW1C	Биты доступны для чтения. Для сброса текущего состояния бита необходимо записать единицу. Запись нуля игнорируется.
RWAC	Регистры с автоматическим сбросом. Запросы к контроллеру сопровождаются установкой битов. Контроллер должен очистить эти биты автоматически, после выполнения необходимых операций. Запись нуля в эти биты игнорируется.
Hwinit	Аппаратно инициализируемые биты. Состояние битов заморожено. После инициализации эти биты доступны только на чтение. Запись в эти биты игнорируется. Изменение этих битов доступно только по средствам изменения конфигурационных регистров и последующего сброса контроллера.
RWRST	Регистры, доступные как на запись, так и на чтение. Но изменения, внесенные в эти регистры, отразятся на работе модуля только после проведения программного сброса с помощью флага Software Reset For All из регистра Software.
Rsvd	Зарезервированные биты. При инициализации устанавливаются в ноль. Запись в эти биты игнорируется.

28.3.1 Регистр SDMA System Address (смещение 000h)

Таблица 28.3. Описание полей регистра SDMA System Address

Номер бита	Поле	Тип	Описание	Сброс
31:0	SDMA System Address / Argument 2	RW	Регистр содержит либо физический адрес системной памяти, использующийся для DMA-транзакций, либо второй аргумент операции Auto CMD23 1) SDMA System Address (Системный адрес SDMA) Этот регистр содержит адрес системной памяти для SDMA передач. Когда Хост Контроллер останавливает SDMA передачу, этот регистр должен указывать на следующую позицию данных. Операции чтения в ходе транзакции могут возвращать неверное значение. Хост Драйвер должен устанавливать этот регистр до начала SDMA транзакции. После остановки SDMA, новый системный адрес следующей позиции данных может быть прочитан из этого регистра. SDMA передача каждый раз ожидает на границе, указываемой в параметре Host SDMA Buffer Boundary , находящемся в Регистре Block Size . Хост Контроллер генерирует DMA прерывания для того чтобы Хост Драйвер мог обновить этот регистр. Хост Драйвер записывает новый системный адрес следующего блока данных в этот регистр. Хост Контроллер перезапускает SDMA передачу когда записан самый старший байт этого регистра (003h). Когда происходит перезапуск SDMA (с помощью команды Resume, либо с помощью установки Continue Request в регистре Block Gap Control) , Хост Контроллер стартует по следующему адресу, сохраненному в регистре SDMA System Address . ADMA не использует этот регистр. 2) Argument 2 Этот регистр используется с командой Auto CMD23, для установки 32-х битного значения количества блоков в аргумент CMD23, пока выполняется Auto CMD23. 32-х битное значение количества блоков может быть использовано, если Auto CMD23 используется с ADMA. Если Auto CMD23 используется без ADMA, доступное значение количества блоков ограничено регистром Block Count . Максимальное количество блоков в этом случае равно 65535 .	0

28.3.2 Регистр Block size (смещение 004h)

Таблица 28.4. Описание полей регистра Block size

Номер бита	Поле	Тип	Описание	Сброс
15	–	Rsvd	Зарезервировано	0
14:12	Host SDMA Buffer Size	RW	Для выполнения долгой DMA пересылки, регистр System Address должен обновляться в течении этой пересылки на каждой границе буфера. Эти биты определяют размер непрерывного буфера в системной памяти. DMA передача должна ожидать на каждой границе буфера, задаваемой битами данного поля. Хост Контроллер генерирует DMA прерывания для того чтобы Хост Драйвер мог обновить регистр System Address . Эти биты должны использоваться, когда параметр DMA Support из регистра Capabilities установлен в 1 и эта функция активна (когда параметр DMA Enable в регистре Transfer Mode установлен в 1) 000b - 4KB(Detects A11 Carry out) 001b - 8KB(Detects A12 Carry out) 010b - 16KB(Detects A13 Carry out) 011b - 32KB(Detects A14 Carry out) 100b - 64KB(Detects A15 Carry out) 101b -128KB(Detects A16 Carry out) 110b - 256KB(Detects A17 Carry out) 111b - 512KB(Detects A18 Carry out)	0
11:0	Transfer Block Size	RW	Этот регистр определяет размер блока для блочных передач данных CMD17, CMD18, CMD24, CMD25 и CMD 53. Он доступен, только когда никакие транзакции не выполняются (например, после остановки транзакции). Операции чтения во время передачи возвращают неверное значение, а операции записи должны игнорироваться. 0000h - No Data Transfer 0001h - 1 Byte 0002h - 2 Bytes 0003h - 3 Bytes 0004h - 4 Bytes . . . . . . 01FFh - 511 Bytes 0200h - 512 Bytes . . . . . . 0800h - 2048 Bytes	0

28.3.3 Регистр Block count (смещение 006h)

Таблица 28.5. Описание полей регистра Block count

Номер бита	Поле	Тип	Описание	Сброс
15:0	Blocks Count for Current Transfer	RW	Это регистр доступен, когда параметр Block Count Enable в регистре Transfer Mode установлен в 1 и имеет силу только для множественных блоковых передач. Хост контроллер уменьшает количество блоков после каждой блочной передачи и останавливается, когда это число достигает нуля. Доступен, только когда никакие транзакции не выполняются (например, после остановки транзакции). Операции чтения во время передачи возвращают неверное значение, а операции записи должны игнорироваться. При сохранении передачи контекста в результате команды Suspend , число еще не переданных блоков может быть определено путем чтения этого регистра. При восстановлении контекста передачи до выдачи команды Resume, Хост Драйвер должен восстановить ранее сохраненное количество блоков. 0000h - Stop Count 0001h - 1 block 0002h - 2 blocks --- --- FFFFh - 65535 blocks	0

28.3.4 Регистр Argument1 (смещение 008h)

Таблица 28.6. Описание полей регистра Argument1

Номер бита	Поле	Тип	Описание	Сброс
31:0	Command Argument 1	RW	Command Argument определяет биты 39-8 формата команды	0

28.3.5 Регистр Transfer mode (смещение 00Ch)

Таблица 28.7. Описание полей регистра Transfer mode

Номер бита	Поле	Тип	Описание	Сброс
15:6	–	Rsvd	Зарезервировано	0
5	Multi / Single Block Select	RW	Это бит включает режим множественной передачи блока. 0 — Единичный Блок 1 — Множество Блоков	0

Номер бита	Поле	Тип	Описание	Сброс
4	Data Transfer Direction Select	RW	Этот бит определяет направление передачи данных. 0 — Запись (от Хоста на карту) 1 — Чтение (от карты к хосту)	0
3:2	Auto CMD Enable	RW	Это поле определяет использование функции автоматической команды. 00b - Auto Command Disabled 01b - Auto CMD12 Enable 10b - Auto CMD23 Enable 11b — Reserved Существует два метода остановки операций чтения и записи множественных блоков. 1) Включение автоматического CMD12 Команды чтения и записи множества блоков нуждаются в CMD12 для остановки операции. Когда это поле установлено в 01b Хост Контроллер автоматически выдает CMD12 когда передан последний блок. Ошибки при автоматическом CMD12 отображаются в регистр Auto CMD Error Status . Хост драйвер не должен устанавливать этот бит, если не предполагается использовать CMD12 2) Включение автоматического CMD23 Когда в этом битовом поле установлено 10b , Хост Контроллер выдает автоматически CMD23 перед выдачей команды, указанной в Command регистре . Требуются следующие условия для использования Auto CMD23 : - поддержка Auto CMD23 (версия Хост Контроллера 3.0 или выше) - карта памяти, которая поддерживает CMD23 (SCR[33]=1) - если используется DMA, должно быть ADMA - только когда выданы CMD18 или CMD25 При записи Command регистра Хост Контроллер выдает сначала CMD23 , а уже затем выдает команду, которая определяется параметром Command Index в регистре Command . 32-х битное значение количества блоков для CMD23 устанавливается регистром SDMA System Address / Argument 2	0
1	Block Count Enable	RW	Этот бит используется для разрешения использования регистра Block count, который актуален для множественной пересылки блоков. Если этот бит установлен в 0, то работа с регистром Block Count запрещена, что может быть полезно при выполнен бесконечной передачи.	0

Номер бита	Поле	Тип	Описание	Сброс
			0 — Запрещено 1 — Разрешено	
0	DMA Enable	RW	DMA может быть разрешено, только если установлен бит DMA Support в регистре Capabilities . Если этот бит установлен в 1, DMA операция должна начаться при записи Хост Драйвером старшего байта в регистр Command (00Fh). 0 — Запрещено 1 — Разрешено	0

Типы передачи определяются по следующей таблице:

Таблица 28.8. Типы передачи

Multi / Single Block Select	Block Count Enable	Block Count	Функция
0	Не важно	Не важно	Единичная передача
1	0	Не важно	Бесконечная передача
1	1	Не ноль	Множественная передача
1	1	ноль	Остановка множественной передачи

28.3.6 Регистр Command (смещение 00Eh)

Таблица 28.9. Описание полей регистра Command

Номер бита	Поле	Тип	Описание	Сброс
15:14	–	Rsvd	Зарезервировано	0
13:8	Command Index	RW	В эти биты должен быть записан номер команды (CMD0-63, ACMD0- 63)	0
	Command Type	RW	Существует три типа специальных команд: Suspend, Resume и Abort. Для всех остальных команд эти биты должны быть установлены в 00b. Команда Suspend При успешном выполнении команды Suspend, Хост Драйвер должен взять управление освободившейся шиной SD и может подавать следующую команду, которая использует линии DAT. Хост контроллер должен установить Read Wait для операции чтения и перестать проверять занятость для операций записи. Цикл прерываний начинается в 4-х битном режиме. Если команда Suspend не удается, Хост Контроллер должен сохранить свое текущее состояние, а Хост Драйвер должен перезапустить передачу. Установкой параметра Continue Request в регистре Block Gap Control. Команда Resume	0

Номер бита	Поле	Тип	Описание	Сброс
			Хост драйвер перезапускает передачу данных путем перезаписи регистров в диапазоне 000-00Dh. Хост-контроллер должен провести проверку на занятость до начала операций записи. Команда *Abort* Если эта команда подается во время выполнения операции чтения, то Хост Контроллер должен остановить чтение в буфер. Если же эта команда подается во время операции записи, , то Хост контроллер должен перестать управлять линиями DAT. После подачи команды Abort , Хост Драйвер должен произвести программный сброс. 00b - Normal 01b - Suspend 10b - Resume 11b - Abort	
	Data Present Select	RW	Если этот бит установлен в 1, то это показывает, что данные готовы и должны быть переданы по линиям DAT. Бит устанавливается в 0 для следующего: 1. Команды, использующие только линии CMD (например, CMD52) 2. Команды, не использующие передачу данных, но использующие сигнал занятости на линии DAT[0] (R1b or R5b ex. CMD38) 3. Команда Resume 0 — данные не готовы 1 — данные готовы	0
	Command Index Check Enable	RW	Если этот бит установлен в 1, то Хост Контроллер будет проверять поле индекса в ответе, чтобы увидеть, то ли самое это значение, что и индекс команды. Если будет несовпадение, то это расценивается как Command Index Error. Если этот бит сброшен в 0, то поле индекса не проверяется. 0 — Запрещено 1 — Разрешено	0
	Command CRC Check Enable	RW	Если этот бит установлен в 1, то Хост Контроллер будет проверять поле CRC в ответе. Если обнаружится ошибка, то это будет расцениваться как Command CRC Error . Если этот бит сброшен в 0, то поле CRC не проверяется. 0 — Запрещено 1 — Разрешено	0
	–	Rsvd	Зарезервировано	0
	Response Type Select	RW	Выбор типа ответа. 00 — без ответа 01 — длина ответа 136 10 — длина ответа 48	0

Номер бита	Поле	Тип	Описание	Сброс
			11 — длина ответа 48 проверка занятости после ответа	

28.3.7 Регистр Response (смещение 010h)

Таблица 28.10. Описание полей регистра Response

Номер бита	Поле	Тип	Описание	Сброс
127:0	Command Response	ROC	R[] относится к диапазону битов в пределах ответных данных, переданных по шине SD, REP[] относится к диапазону битов в пределах регистра Response	0

В следующей таблице приведена зависимость между некоторыми параметрами и названиями типов ответов

Таблица 28.11. Зависимость между параметрами и названиями типов ответов

Response Type	Index Check Enable	CRC Check Enable	Название типа ответа
00	0	0	Без ответа
01	0	1	R2
10	0	0	R3, R4
10	1	1	R1, R6, R5, R7
11	1	1	R1b, R5b

Следующая таблица описывает соответствие ответных команд от шины SD к данному регистру и каждого типа ответов. R[] относится к диапазону битов в пределах ответных данных, переданных по шине SD, REP[] относится к диапазону битов в пределах регистра Response.

Таблица 28.12. Соответствие ответных команд к регистру

Тип ответа	Значение ответа	Поле ответа	Регистр Response
R1, R1b (обычный ответ)	Статус карты	R[39:8]	REP[31:0]
R1b (Автоматический CMD12 ответ)	Статус карты для автоматического CMD12	R[39:8]	REP[127:96]
R1 (Автоматический CMD23 ответ)	Статус карты для автоматического CMD23	R[39:8]	REP[127:96]
R2 (CID, CSD регистр)	CID or CSD reg. incl.	R[127:8]	REP[119:0]
R3 (OCR регистр)	OCR регистр для памяти	R[39:8]	REP[31:0]
R4 (OCR регистр)	OCR регистр для I/O и проч.	R[39:8]	REP[31:0]
R5, R5b	SDIO ответ	R[39:8]	REP[31:0]
R6 (Публичный RCA ответ)	Новый публичный RCA[31:16]	R[39:8]	REP[31:0]

28.3.8 Регистр Buffer data port (смещение 020h)

Таблица 28.13. Описание полей регистра Buffer data port

Номер бита	Поле	Тип	Описание	Сброс
31:0	Buffer Data	RW	Буфер Хост контроллера может быть доступен через данный 32-х битный регистр Data Port	X

28.3.9 Регистр Present state (смещение 024h)

Таблица 28.14. Описание полей регистра Present state

Номер бита	Поле	Тип	Описание	Сброс
31:29	–	Rsvd	Зарезервировано	0
28:25	DAT[7:4] Line Signal Level	RO	Этот статус используется для проверки уровня на линиях DAT чтобы была возможность восстановления после ошибок и для отладки. D28 - DAT[7] D27 - DAT[6] D26 - DAT[5] D25 - DAT[4]	4'hf
24	CMD Line Signal Level	RO	Этот статус используется для проверки уровня на линиях CMD, чтобы была возможность восстановления после ошибок и для отладки.	1'b1
23:20	DAT[3:0] Line Signal Level	RO	Этот статус используется для проверки уровня на линиях DAT, чтобы была возможность восстановления после ошибок и для отладки. Это особенно полезно для детектирования уровня сигнала занятости от DAT[0] D23 - DAT[3] D22 - DAT[2] D21 - DAT[1] D20 - DAT[0]	4'hf
19	Write Protect Switch Pin Level	RO	Этот бит отображает состояние пина SDWP# 0 — Запись запрещена (SDWP# = 1) 1 — Запись разрешена (SDWP# = 0)	0
18	Card Detect Pin Level	RO	Этот бит отображает инвертированное значение уровня с пина SDCD# 0 — Карта отсутствует (SDCD# = 1) 1 — Карта присутствует (SDCD# = 0)	0
17	Card State Stable	RO	Данный бит используется для тестирования. Если этот бит равен 0 , значит предыдущий параметр Card Detect Pin Level не стабилен, если 1, то стабилен. Параметр Software Reset For All в регистре Software Reset не оказывает влияния на данный бит. 0 — Reset или процесс устранения дребезга 1 — Карта отсутствует или присутствует	0

Номер бита	Поле	Тип	Описание	Сброс
16	Card Inserted	RO	Данный бит показывает, была ли вставлена карта. Изменение из 0 в 1 генерирует прерывание Card Insertion в регистре Normal Interrupt Status , а изменение из 1 в 0 — прерывание Card Removal Interrupt в том же регистре. Параметр Software Reset For All в регистре Software Reset не оказывает влияния на данный бит. Если карта извлечена, пока подано напряжение и работает клок, то Хост Контроллер должен очистить параметр SD Bus Power в регистре Power Control и параметр SD Clock Enable в регистре Clock control . В дополнение Хост Драйвер должен произвести сброс Хост контроллера с использованием параметра Software Reset For All из регистра Software. Наличие карты определяется независимо от питания шины SD 0 — Reset или процесс устранения дребезга или карта отсутствует 1 — Карта вставлена	0
15:12	–	Rsvd	Зарезервировано	0
11	Buffer Read Enable	ROC	Этот статус используется для non-DMA передач чтения. Этот флаг показывает, что валидные данные находятся в буфере на стороне хоста. Если этот бит установлен в 1, то данные для чтеня находятся в буфере. Изменение из 1 в 0 происходит когда весь блок данных прочитан из буфера. Изменение этого бита из 0 в 1 происходит, когда весь блок данных готов и находится в буфере и генерируется прерывание Buffer Read Ready . 0 — Чтение запрещено 1 — Чтение разрешено	0
10	Buffer Write Enable	ROC	Этот статус используется для non-DMA передач чтения. Этот флаг показывает, доступно ли место в буфере для записываемых данных. Если этот бит установлен в 1, то данные могут быть записаны в буфер. Изменение этого бита из 1 в 0 происходит, когда весь блок данных записан в буфер. Изменение из 0 в 1 происходит, когда верхняя часть блока данных может быть записана в буфер и генерируется прерывание Buffer Write Ready . 0 — Запись запрещена 1 — Запись разрешена	0
9	Read Transfer Active	ROC	Этот статусный бит используется для определения окончания операции чтения.	0

Номер бита	Поле	Тип	Описание	Сброс
			Он может устанавливаться в 1 при выполнении любого из следующих условий: - после последнего бита команды на чтение - когда записывается 1 в Request в регистре Block Gap Control , чтобы перезапустить операцию чтения Этот бит сбрасывается в 0, при выполнении одного из следующих условий: - когда последний блок данных, определяющий длину блока, передан в систему. - когда все валидные блоки данных переданы в систему и никаких текущих передач не происходит, в результате чего Stop At Block Gap Request устанавливается в 1. Когда этот бит сбрасывается в 0, генерируется прерывание о прекращении передачи. 1 — Передача данных 0 — Нет валидных данных	
8	Write Transfer Active	ROC	Этот статусный бит показывает, активна ли посылка на запись. Если данный бит установлен в 0, это означает что в Хост Контроллере нет валидной даты для записи. Этот бит устанавливается в следующих случаях: - после последнего бита команды на чтение - когда записана 1 в поле Continue Request в регистре Block Gap Control для перезапуска операции записи. Этот бит очищается в следующих случаях: - после получения статуса CRC последнего блока данных, определяющего число передач -после получения статуса CRC любого блока, на котором передача данных собирается быть прервана с помощью параметра Stop At Block Gap Request. В течение передачи на запись, когда этот бит установлен, может быть сгенерировано прерывание Block Gap Event , как результат установки Stop At Block Gap Request . 1 — Передача данных 0 — Нет валидных данных	0
7:4	–	Rsvd	Зарезервировано	0
3	Re-Tuning Request	ROC	Установкой этого бита Хост контроллер может запросить Хост Драйвер о выполнении подстроечной последовательности , когда окно данных сдвигается из-за температурного дрейфа и точка выборки уже не имеет требуемого запаса для получения корректных данных.	0

Номер бита	Поле	Тип	Описание	Сброс
			Этот бит очищается, когда запускается команда Execute Tuning из регистра Host Control 2 . При изменение этого бита из 0 в 1 генерируется Re-Tuning Event. (подробнее в описании регистра Normal Interrupt). Этот бит не устанавливается, если параметр Sampling Clock Select из регистра Host Control 2 сброшен в 0 (используется фиксированная частота выборки). 1 — Частота выборки нуждается в подстройке 0 — Фиксированная или удовлетворительная частота выборки	
2	DAT Line Active	ROC	Этот бит показывает, используется ли какая-нибудь из линий DAT	0
1	Command Inhibit (DAT)	ROC	Этот статусный бит выставляется либо когда активна одна из линий DAT или когда параметр Read Transfer Active из данного регистра установлен в 1. Когда этот бит сброшен в 0, это показывает что Хост Контроллер может выполнить следующую команду SD . Команды с сигналом занятости связаны с параметром Command Inhibit (DAT) (ex. R1b, R5b type). При изменение этого бита из 1 в 0 генерируется прерывание Transfer Complete в регистре Normal interrupt Status . **Примечание**. SD Хост Драйвер может сохранять регистры из диапазона 000-00Dh для приостановки передачи после изменения этого бита из 1 в 0. 1 — невозможно выполнить команду, которая использовала бы линии DAT 0 — выполнение команды, использующей линии DAT, возможно.	0
0	Command Inhibit (CMD)	ROC	Если этот бит сброшен в 0 , то это показывает что линии CMD не используются и Хост контроллер может выполнить команду SD использующие эти линии. Этот бит устанавливается сразу после записи в регистр Command (00Fh) . Это бит сбрасывается, когда получена ответная команда. Даже если параметр Command Inhibit (DAT) установлен в 1, то команды использующие линии CMD могут все равно быть выполнены, если этот бит сброшен. Изменение из 1 в 0 генерирует прерывание Command Complete в регистре Normal Interrupt Status . Если Хост контроллер не может выполнить команду, потому что появляется ошибка Command Conflict Error или Command Not Issued By Auto CMD12	0

Номер бита	Поле	Тип	Описание	Сброс
			Error , то этот бит остается в 1 и Command Complete не устанавливается. Статус исполнения Auto CMD12 не читается из этого бита. Так как каждая из команд Auto CMD12 и Auto CMD23 подразумевают 2 ответа, то данный бит не сбрасывается при ответе на CMD12 или CMD23, но сбрасывается при ответе на последующую команду чтения/записи. Статус выполнения Auto CMD12 не доступен из чтения этого бита. Поэтому если рассматриваемый бит сбрасывается в течении операции Auto CMD12, то это означает, что Хост Контроллер смог выполнить две команды: CMD12 и команду, установленную в регистре Command.	

28.3.10 Регистр Host control 1 (смещение 028h)

Таблица 28.15. Описание полей регистра Host control 1

Номер бита	Поле	Тип	Описание	Сброс
7	Card detect signal Selection	RW	Этим битов выбирается источник наличия карты. 1 — Для детектирования карты используется тестовый уровень 0 — используется пин SDCD# (в нормальном режиме)	0
6	Card Detect Test Level	RW	Этот бит доступен пока параметр Card Detect Signal Selection установлен в 1 и он показывает вставлена ли карта, или нет. Генерирует прерывание (карта вставлена или извлечена), когда установлен соответствующий бит в регистре Normal Interrupt Status Enable. 1 — Карта вставлена 0 — Карта отсутствует	0
5	Extended Data Transfer Width	RW	Этот бит контролирует 8-ми битный режим ширины шины. Поддержка этой функции указывается в параметре Extended Media Bus Support из регистра Capabilities . Если устройство поддерживает 8-ми битную шину, то этот бит может быть установлен в 1. Если же данный бит сброшен в 0, то ширина шины контролируется параметром Data Transfer Width из регистра Host Control 1 . Этот бит не имеет силы, когда на шине находится несколько устройств (в этом случае в параметр Slot Type регистра Capabilities установлен в 10b). 1 — 8-ми битная ширина шины 0 — ширина шины определяется Data Transfer Width	0

Номер бита	Поле	Тип	Описание	Сброс
4:3	DMA Select	RW	Может быть выбран один из режимов, поддерживающих DMA. Хост драйвер должен проверить поддержку DMA режимов, обратившись к регистру Capabilities . 00 — выбран SDMA 01 — выбран 32-bit Address ADMA1 10 — выбран 32-bit Address ADMA2 11 — выбран 64-bit Address ADMA2	0
2	High Speed Enable	RW	Этот бит опциональный. Перед установкой данного бита Хост Драйвер должен быть проверить параметр High Speed Support из регистра Capabilities . Если этот бит сброшен в 0 (по умолчанию) Хост Контроллер управляет шинами CMD и DAT по заднему фронту SD клока (до 25 Mhz для SD/20MHz для MMC). Если же этот бит установлен в 1, то Хост Контроллер управляет шинами CMD и DAT по переднему фронту SD клока (до 50 Mhz для SD/52MHz для MMC)/208Mhz (для SD3.0) . Если параметр Preset Value Enable из регистра Host Control 2 установлен в 1, то Хост Контроллер должен сбросить SD Clock Enable перед изменением этого поля для того, чтобы избежать появления кличей. После установки этого поля, Хост Драйвер выставляет SD Clock Enable вновь. 1 - High Speed Mode 0 - Normal Speed Mode	0
1	Data Transfer Width (SD1 or SD4)	RW	Этот бит определяет ширину шины данных Хост Контроллера. Хост драйвер должен выбрать этот параметр в соответствии с шириной шины данных SD карты. 1 — 4-х битный режим 0 — 1 битный режим	0
0	LED Control	RW	Этот бит используется для предупреждения пользователя, чтобы он не извлекал SD карту, пока к ней осуществляется доступ. Если ПО намерено выполнить множество SD команд, этот бит может быть установлен на протяжении всех транзакций. Не обязательно изменять его для каждой транзакции. 1 - LED on 0 - LED off	0

28.3.11 Регистр Power control (смещение 029h)

Таблица 28.16. Описание полей регистра Power control

Номер бита	Поле	Тип	Описание	Сброс
7:5	–	Rsvd	Зарезервировано	0
4	Hardware reset	RW	Сигнал аппаратного сброса, генерируемый для eMMC карты когда данный бит установлен в 1. Активный уровень сигнала сброса для eMMC	0

Номер бита	Поле	Тип	Описание	Сброс
			карты — ноль. Запись 0 в данный бит, отпускает пин аппаратного сброса.	
3:1	SD Bus Voltage Select	RW	Установкой данного бита Хост контроллер выбирает уровень напряжения на SD карте. Перед установкой данного поля, Хост Драйвер должен проверить параметр Voltage Support из регистра Capabilities . Если выбрано неподдерживаемое напряжение, система не должна поставлять питание на шину SD. 111b - 3.3 Flattop.) 110b - 3.0 V(Typ.) 101b - 1.8 V(Typ.) 100b - 000b - Зарезервировано	0
0	SD Bus Power	RW	Перед установкой этого бита, SD хост драйвер должен установить SD Bus Voltage Select. Если Хост Контроллер выявит отсутствие карты, то этот бит должен быть очищен. 1 — Питание есть 0 — Питание отсутствует	0

28.3.12 Регистр Block gap control (смещение 02Ah)

Таблица 28.17. Описание полей регистра Block gap control

Номер бита	Поле	Тип	Описание	Сброс
7	boot_ack_chk	RW	Проверить подтверждение загрузки в соответствующих операциях 1 — ожидать подтверждения загрузки с eMMC карты 0 — не ожидать подтверждения загрузки с eMMC карты	1
6	alt_boot_en	RW	Начать обращение к коду загрузки в альтернативном режиме 1 — начать обращение к альтернативному режиму загрузки 0 — остановить обращение к альтернативному режиму загрузки	0
5	BOOT_EN	RW	Обращение к коду загрузки 1 — начать обращение к коду загрузки 2 — остановить обращение к коду загрузки	0
4	SPI_MODE	RW	Бит разрешения режима SPI 1 — режим SPI 0 — режим SD	0
3	Interrupt At Block Gap	RW	Этот бит валиден только в 4-х битном режиме SDIO карты и определяет точку выборки в цикле прерывания. Установка в 1 разрешает детекцию прерываний в промежутках между блоками во время множественной передачи блоков. Если SD карта не может генерировать прерывание во время множественной блоковой передаче, то этот бит должен быть сброшен в 0. Когда Хост Драйвер обнаруживает, что SD карта вставлена,	0

Номер бита	Поле	Тип	Описание	Сброс
			то он должен установить этот бит в соответствии с CCCR карты SDIO .	
2	Read Wait Control	RW	Функция ожидания чтения (Read Wait) опциональна для SDIO карт. Если карта поддерживает эту функцию, необходимо установить этот бит чтобы разрешить использовать протокол ожидания чтения для остановки чтения данных по линии DAT[2] . В противном случае Хост Контроллер вынужден останавливать SD клок для ожидания данных на чтение, что ограничивает формирование команд. Когда Хост Драйвер обнаруживает, что SD карта вставлена, то он должен установить этот бит в соответствии с CCCR карты SDIO . Если карта не поддерживает функцию ожидания чтения, этот бит никогда не должен устанавливаться в 1, иначе на шине DAT может произойти конфликт. В то время, когда данный бит сброшен в 0, не поддерживаются команды Suspend / Resume . 1 — Разрешить ожидание чтения 0 — Запретить ожидание чтения	0
1	Continue Request	RWAC	Этот бит используется для того чтобы перезапустить транзакцию, которая может быть остановлена с помощью Stop At Block Gap Request. Для отмены остановки между блоками необходимо сбросить Stop At block Gap Request в 0 и установить данный бит для перезапуска транзакции. Хост Контроллер автоматически очищает этот бит в одном из следующих случаев: 1) В случае операции чтения, когда DAT Line Active изменяется из 0 в 1, как результат перезапуска операции чтения. 2) В случае операции записи, когда Write Transfer Active изменяется из 0 в 1, как результат перезапуска операции записи. Следовательно, сбрасывать данный бит Хост Драйверу нет необходимости. Если параметр Stop At Block Gap Request установлен в 1, любая запись в данный бит игнорируется 1 - Перезапустить 0 - Игнорировать	0
0	Stop At Block Gap Request	RW	Этот бит используется для того чтобы остановить выполнение транзакции в следующем промежутке между блоками для non-DMA, SDMA и ADMA транзакций.	0
			До тех пор, пока параметр Transfer Complete (из регистра Normal Interrupt Status) не будет установлен в 1, что означает завершение передачи, Хост Драйвер должен оставлять этот бит установленным в 1. Сброс обоих параметров Stop At Block Gap Request и Continue Request не должен вызывать перезагрузку передачи. Read Wait используется для остановки транзакции	

Номер бита	Поле	Тип	Описание	Сброс
			чтения в межблочном интервале. Хост контроллер должен учитывать Stop At Block Gap Request для транзакций записи, но для транзакций на чтение требуется поддержка SD картой функционала Read Wait. Следовательно, Хост Драйвер не должен устанавливать этот бит в процессе транзакций чтения, если SD карта не поддерживает Read Wait , и должен установить Read Wait Control в 1. В случае транзакций на запись, в которых Хост Драйвер записывает данные в регистр Buffer Data Port, Хост драйвер должен устанавливать этот бит после того как блок данных будет записан. Если этот бит установлен в 1, Хост Драйвер не должен писать данные в регистр Buffer Data Port. Этот бит влияет на Read Transfer Active, Write Transfer Active, DAT Line Active and Command Inhibit (DAT) в регистре Present State . 1 — Остановка 0 — Передача	

28.3.13 Регистр Wake-up control (смещение 02Bh)

Таблица 28.18. Описание полей регистра Wake-up control

Номер бита	Поле	Тип	Описание	Сброс
7:3	–	Rsvd	Зарезервировано	0
2	Wakeup Event Enable On SD Card Removal	RW	Этот бит разрешает событие wakeup при установке бита Card Removal в регистре Normal Interrupt Status . FN_WUS (Wake up Support) в CIS не влияет на этот бит. 1 — Разрешено 0 — Запрещено	0
1	Wakeup Event Enable On SD Card Insertion	RW	Этот бит разрешает событие wakeup при установке бита Card Insertion в регистре Normal Interrupt Status. FN_WUS (Wake up Support) в CIS не влияет на этот бит. 1 — Разрешено 0 — Запрещено	0
0	Wakeup Event Enable On Card Interrupt	RW	Этот бит разрешает событие wakeup при установке бита Card Interrupt в регистре Normal Interrupt Status . Этот бит может быть может быть установлен в 1 если FN_WUS (Wake up Support) в CIS установлен в 1. 1 — Разрешено 0 — Запрещено	0

28.3.14 Регистр Clock control (смещение 02Ch)

Во время инициализации Хост Контроллера, Хост Драйвер должен установить SDCLK Frequency Select в соответствии с регистром Capabilities.

Таблица 28.19. Описание полей регистра Clock control

Номер бита	Поле	Тип	Описание	Сброс
15:8	SDCLK Frequency Select	RW	Этот регистр используется для выбора частоты на выходе SDCLK . Эта частота не задается на прямую, этот регистр хранит значение делителя частоты Base Clock Frequency For SD из регистра Capabilities . Доступны только следующие значения: **(1)** 8-bit Divided Clock Mode 80h — исходный клок делится на 256 40h — исходный клок делится на 128 20h — исходный клок делится на 64 10h — исходный клок делится на 32 08h — исходный клок делится на 16 04h — исходный клок делится на 8 02h — исходный клок делится на 4 01h — исходный клок делится на 2 00h — исходный клок (10MHz-63MHz) Установка 00h определяет наивысшую частоту SD клока. При установке нескольких бит в регистре, значение делителя будет определяться старшим битом. Но несколько бит одновременно в этом регистре не должно устанавливаться. Два значения делителей по умолчанию вычисляются по частоте, которая определена параметром Base Clock Frequency For SD Clock в регистре Capabilities . 1) 25 МГц значение делителя 2) 400 КГц значение делителя Частота SDCLK устанавливается исходя из следующей формулы: Clock Frequency = (Base clock) / divisor. Таким образом, выбор наименьшего значения делителя даст значение частоты меньшей или равной значению исходной частоты. Максимальная частота для SD = 50 МГц (исходная) Максимальная частота для MMC = 50 МГц (исходная) Минимальная частота = 195.3125Khz (50Mhz / 256), также и для MMC. **(2)** 10-bit Divided Clock Mode Хост Контроллер версии 3.00 поддерживает этот обязательный режим вместо 8-bit Divided Clock Mode . Длина делителя расширена до 10 бит и должны поддерживаться все значения делителя. 3FFh --1/2046 Divided Clock N -------1/2N Divided Clock (Duty 50%) 002h -- 1/4 Divided Clock 001h ---1/2 Divided Clock 000h --- Base Clock (10MHz-254MHz)	0
7:6	Upper Bits of SDCLK Frequency Select	RW	Это биты 09-08 делителя клока SDCLK Frequency Select	0
5	Clock Generator Select	RW	Этот бит используется для выбора режима генерации частоты в SDCLK Frequency Select.	0

Номер бита	Поле	Тип	Описание	Сброс
			Если используется режим Programmable Clock Mode (при ненулевом значении параметра Clock Multiplier регистре Capabilities), то этот бит доступен как на чтение, так и на запись. Если не поддерживается, то бит доступен только на чтение и читается 0. Этот бит зависит от значения Preset Value Enable в регистре Host Control 2 . Если Preset Value Enable = 0, то этот бит устанавливается Хост Драйвером. Если Preset Value Enable = 1, то этот бит автоматически устанавливается в соответствии со значениями регистров Preset Value . 1 Programmable Clock Mode 0 Divided Clock Mode	
4:3	–	Rsvd	Зарезервировано	0
2	SD Clock Enable	RW	Хост Контроллер должен останавливать тактовую частоту SDCLK , когда в этот бит записан 0. Параметр SDCLK Frequency Select может быть изменен, когда этот бит установлен в 0. Затем Хост Контроллер должен установить ту же частоту, что была до тех пор, как SDCLK была остановлена. Если Хост Контроллер выявит отсутствие карты, то этот бит должен быть очищен. 1 — Разрешено 0 — Запрещено	0
1	Internal Clock Stable	ROC	Этот бит устанавливается в 1, когда SD клок становится стабильным после записи 1 в Internal Clock Enable из этого регистра. SD Хост Драйвер должен подождать с установкой SD Clock Enable , до тех пор пока не установится данный бит.	0
0	Internal Clock Enable	RW	Этот бит устанавливается в 0, когда Хост Драйвер не использует Хост Контроллер или когда Хост Контроллер ожидает событие wakeup . Хост Контроллер должен остановить свою внутреннюю частоту чтобы перейти в состояние с более низким энергопотреблением. Но регистры в это время все еще доступны на запись и чтение. Тактовая частота включается, когда этот бит устанавливается в 1. Когда частота становится стабильной, Хост Контроллер должен установить параметр Internal Clock Stable в 1. Этот бит не должен влиять на детектирование наличия карты 1 — Частота включена 2 — Частота остановлена.	0

28.3.15 Регистр Timeout control (смещение 02Eh)

Таблица 28.20. Описание полей регистра Timeout control

Номер бита	Поле	Тип	Описание	Сброс
7:4	–	Rsvd	Зарезервировано	0
3:0	Data Timeout Counter Value	RW	Это значение определяет интервал, по которому детектируется time-out на DAT-линиях. Информация о факторах, вызывающих time-out описывается параметром Data Time-out Error в регистре Error Interrupt Status . Частота для определения time-out будет сгенерирована делением sdclockTMCLK на значение этого регистра. Когда этот регистр установлен, предотвращение нежелательных time-out событий происходит при помощи очистки параметра Data Time-out Error Status Enable в регистре Error Interrupt Status Enable . 1111 - Reserved 1110 - TMCLK * 2^27 ---------- ---------- 0001 - TMCLK * 2^14 0000 - TMCLK * 2^13	0

28.3.16 Регистр Software reset (смещение 02Fh)

Таблица 28.21. Описание полей регистра Software reset

Номер бита	Поле	Тип	Описание	Сброс
7:3	–	Rsvd	Зарезервировано	0
2	Software Reset for DAT Line	RWAC	Сбрасывается только часть цепи данных. Следующие регистры и биты очищаются этим битом: Buffer Data Port Register Буфер очищен и проинициализирован Present State register Buffer read Enable Buffer write Enable Read Transfer Active Write Transfer Active DAT Line Active Command Inhibit (DAT) Block Gap Control register Continue Request Stop At Block Gap Request Normal Interrupt Status register Buffer Read Ready Buffer Write Ready Block Gap Event Transfer Complete 1 — Сброс	0

Номер бита	Поле	Тип	Описание	Сброс
			0 — Работает	
1	Software Reset for CMD Line	RWAC	Сбрасывается только часть командной цепи. Следующие регистры и биты очищаются этим битом: Present State register Command Inhibit (CMD) Normal Interrupt Status register Command Complete 1 — Сброс 0 — Работает	0
0	Software Reset for All	RWAC	Этот сброс влияет на весь Хост Контроллер за исключением цепи детектирования карты. Биты регистров с типом ROC, RW, RW1C, RWAC сбрасываются в 0. Во время инициализации Хост Драйвер должен установить этот бит в 1 для сброса Хост Контроллера. Хост контроллер должен самостоятельно сбросить этот бит в 0, когда регистры Capabilities становятся валидными и Хост Драйвер способен их прочитать. Использование Software Reset For All не оказывает влияния на значение регистров Capabilities . Если этот бит установлен в 1, SD карта производит внутренний сброс и должна быть проинициализирована Хост Драйвером 1 — Сброс 0 — Работает	0

28.3.17 Регистр Normal interrupt status (смещение 030h)

Таблица 28.22. Описание полей регистра Normal interrupt status

Номер бита	Поле	Тип	Описание	Сброс
15	Error Interrupt	ROC	Если в регистре Error Interrupt Status установлен какой-либо бит, то бит Error Interrupt устанавливается в 1 автоматически. Следовательно, Хост Драйвер может провести тест на наличие ошибок, прочитав этот бит в первую очередь. 0 — Нет ошибок 1 — Имеются ошибки	0
14	Boot terminate Interrupt	RW1C	Этот статусный бит если операция загрузки прервана 0 — операция загрузки не прерывалась 1 — операция загрузки прервана	0
13	Boot ack rcv	RW1C	Этот статусный бит устанавливается, если от устройства принято подтверждение загрузки 0 — подтверждение загрузки не принято 1 — принято подтверждение загрузки	0
12	Re-Tuning Event	ROC	Этот статусный бит устанавливается, если параметр Re-Tuning Request в регистре Present State изменяется из 0 в 1.	0

Номер бита	Поле	Тип	Описание	Сброс
			Хост Контроллер запрашивает Хост Контроллер о проведении процедуры подстройки для следующей передачи данных. Текущая передача данных (не большое число блоков) может быть завершена без подстройки. 1 — Должна быть произведена процедура подстройки 0 — Подстройка не требуется	
11	int_c	ROC	Этот статусный бит установлен, если разрешено INT_C и на пине INT_C# низкий уровень. Запись 1 в этот бит не очищает его. Он очищается после исчезновения фактора, вызвавшего прерывание INT_C	0
10	int_b	ROC	Этот статусный бит установлен, если разрешено INT_B и на пине INT_B# низкий уровень. Запись 1 в этот бит не очищает его. Он очищается после исчезновения фактора, вызвавшего прерывание INT_B	0
9	int_a	ROC	Этот статусный бит установлен, если разрешено INT_A и на пине INT_A# низкий уровень. Запись 1 в этот бит не очищает его. Он очищается после исчезновения фактора, вызвавшего прерывание INT_A	0
8	Card Interrupt	ROC	Запись 1 в этот бит не очищает его. Он очищается после исчезновения фактора, вызвавшего прерывание SD карты . В 1-битном режиме Хост Контроллер должен детектировать Card Interrupt без SD клока для того чтобы иметь возможность поддерживать функционал wakeup. В 4-х битном режиме сигнал прерывания от карты детектируется в течении цикла прерывания, поэтому существует некоторая задержка между сигналом прерывания с карты и прерыванием в Хост Системе . Когда этот статусный бит поднят и Хост Драйверу нужно запустить обработку прерываний, то чтобы очистить этот, нужно сбросить в 0 бит Card Interrupt Status Enable в регистре Normal Interrupt Status Enable . После завершения обработки прерывания необходимо установить Card Interrupt Status Enable в 1, и снова возобновить ожидание сигнала прерывания. Детектирование прерывания по линии DAT[1] поддерживается при условии, что используется только одна карта на слот. В случае использовании общей шины, для детектирования прерываний используются пины прерываний. 0 — Не было прерывания от карты 1 — Обнаружено прерывание от карты	0
7	Card Removal	RW1C	Этот статусный бит устанавливается, если параметр Card Inserted в регистре Present State изменяется из 1 в 0. Чтобы очистить этот бит, Хост Драйвер должен записать в него 1, в этом	0

Номер бита	Поле	Тип	Описание	Сброс
			случае статус Card Inserted в регистре Present State должен согласованно измениться. Поскольку детектирование карты может произойти в то время, когда Хост Драйвер очищает этот бит, событие прерывания может быть не сгенерировано. 0 — Состояние карты стабильно или процесс устранения дребезга 1 — Карта извлечена	
6	Card Insertion	RW1C	Этот статусный бит установлен, если параметр Card Inserted регистра Present State изменился с 0 на 1. Для сброса этого бита, Хост Драйвер должен записать в него 1. Параметр Card Inserted регистра Present State в этом случае также должен измениться соответствующим образом. Поскольку детектирование карты может произойти в то время, когда Хост Драйвер очищает этот бит, событие прерывания может быть не сгенерировано. 0 — Состояние карты стабильно или процесс устранения дребезга 1 — Карта вставлена	0
5	Buffer Read Ready	RW1C	Этот бит выставляется, когда параметр Buffer Read Enable изменяется из 0 в 1. Buffer Read Ready выставляется каждый раз при выполнении CMD19 в процедуре подстройки. 0 — Буфер для чтения не готов 1 — Буфер для чтения готов	0
4	Buffer Write Ready	RW1C	Этот бит выставляется, когда параметр Buffer Write Enable изменяется из 0 в 1. 0 — Буфер для записи не готов 1 — Буфер для записи готов	0
3	DMA Interrupt	RW1C	Этот бит устанавливается, если Хост Контроллер обнаруживает появление Host DMA Buffer Boundary в регистре Block Size 0 — Нет прерывания от DMA 1 — Сгенерировано прерывание от DMA	0
2	Block Gap Event	RW1C	Этот бит выставляется, при появлении Stop At Block Gap Request в регистре Block Gap Control Операция чтения: Этот бит выставляется по заднему фронту сигнала DAT Line Active Status (Когда транзакция остановлена во время пересылки по шине SD, Read Wait должна поддерживать этот функционал.) Операции записи: Этот бит выставляется по заднему фронту сигнала Write Transfer Active Status (после получения статуса CRC во время пересылки по шине SD). 0 — События Block Gap не было 1 — Передача остановлена при Block Gap	0

Номер бита	Поле	Тип	Описание	Сброс
1	Transfer Complete	RW1C	Этот бит устанавливается, когда операция чтение/запись завершена. Операция чтения: Этот бит устанавливается по заднему фронту изменения сигнала Read Transfer Active Status . Существует 2 случая, при которых генерируется прерывание. Первый — когда завершена передача данных установленной длины. (После последних данных, прочитанных Хост Системой). Второй случай, когда данный были остановлены в паузе между блоками, и передача завершена путем установки флага Stop At Block Gap Request в регистре Block Gap Control. (После валидного прочтения данных Хост Системой) . Операция записи: Этот бит устанавливается по заднему фронту параметра DAT Line Active Status . Существует также 2 случая, когда генерируется это прерывание в режиме записи. Первый — когда последние данные установленной длины записаны на карту и отпущен сигнал Busy. Второй случай, когда данный были остановлены в паузе между блоками, и передача завершена путем установки флага Stop At Block Gap Request в регистре Block Gap Control. (После валидной записи данных на SD карту и отпускания сигнала Busy.) **Примечани.** Флаг Transfer Complete имеет более высокий приоритет чем флаг Data Time-out Error. Если оба эти бита установлены одновременно, то считается что передача завершена. **Примечание.** Во время выполнения процедуры подстройки (флаг Execute Tuning установлен в 1), бит Transfer Complete не устанавливается в 1. 0 — передача данных не завершена 1 — передача данных завершена	0
0	Command Complete	RW1C	Этот бит устанавливается, когда мы получаем последний бит в командном ответе (кроме CMD12 и CMD23) . **Примечание**. Флаг Command Time-out Error имеет более высокий приоритет , чем Command Complete. Если оба флага установлены одновременно, то можно считать, что ответ не был принят правильно. 0 — Команда не завершена 1 — Команда завершена.	0

Таблица 28.23. Соотношения между transfer complete и data timeout error

Transfer Complete	Data Timeout Error	Значение статуса
0	0	Прерывание от другого фактора
0	1	Во время передачи произошел Timeout
1	Не имеет значения	Передача данных завершена

Таблица 28.24. Соотношения между command complete и command timeout error

Command Complete	Command Timeout Error	Значение статуса
0	0	Прерывание от другого фактора
Не имеет значения	1	Ответ не был получен в течении 64 циклов SDCLK
1	0	Ответ получен

28.3.18 Регистр Error interrupt status (смещение 032h)

Статусы, определенные в этом регистре, могут быть разрешены в регистре Error Interrupt Status Enable, но не регистром Error Interrupt Signal Enable . Прерывание генерируется, когда Error Interrupt Signal Enable разрешен и хотя бы один статус установлен. Запись 1 очищает бит, а запись 0 игнорируется. При записи регистра может быть очищено более одного статуса.

Таблица 28.25. Описание полей регистра Error interrupt status

Номер бита	Поле	Тип	Описание	Сброс
15:13	Vendor Specific Error Status	Rsvd	Зарезервировано	0
12	Target Response error	RW1C	Происходит, когда детектируется ERROR во время m_hresp (dma транзакция) 1 — Ошибка 0 — Нет ошибки	0
11:10	–	Rsvd	Зарезервировано	0
9	ADMA Error	RW1C	Этот бит устанавливается, когда Хост Контроллер обнаруживает ошибки во время передачи данных, на основе ADMA. Значение возникшей ошибки ADMA сохраняется в регистре ADMA Error Status . 1 — Ошибка 0 — Нет ошибки	0
8	Auto CMD Error	RW1C	Auto CMD12 и Auto CMD23 используют этот статус ошибки. Этот бит устанавливается когда детектируется, что один из битов (D00-D04) регистра Auto CMD Error Status изменяется из 0 в 1. В случае Auto CMD12 данный бит выставляется не только тогда, когда происходит соответствующая ошибка, но и тогда, когда Auto CMD12 не выполняется в связи с ошибкой выполнения предыдущей команды. 1 — Ошибка 0 — Нет ошибки	0
7	Current Limit Error	RW1C	Установкой флага SD Bus Power в регистре Power Control Хост Контроллер запрашивает питание для шины SD. Если Хост Контроллер поддерживает Current Limit Function, то можно защитить устройство от неразрешенных карт отключением питания для карты. В этом случае бит показывает статус неисправности. Наличие 1	0

Номер бита	Поле	Тип	Описание	Сброс
			означает, Хост Контролер не подает питание SD на карту из-за наличия некоторых неисправностей. Наличие 0 означает, что Хост Контроллер подает питание и никаких ошибок не обнаружено. Этот бит должен быть всегда установлен в 0, если Хост Контроллер не использует эту функцию. 1 — Неисправность в подаче питания 0 — Нет ошибки	
6	Data End Bit Error	RW1C	Происходит, когда обнаруживается 0 в последнем бите читаемых данных, которые используются линиями DAT или в последнем бите CRC статуса. 1 — Ошибка 0 — Нет ошибки	0
5	Data CRC Error	RW1C	Происходит, когда обнаруживается ошибка CRC при чтении данных, используя линии DAT, или когда обнаруживается, что значение Write CRC Status отлично от «010». 1 — Ошибка 0 — Нет ошибки	0
4	Data Timeout Error	RW1C	Происходит, когда обнаруживается одно из следующих timeout -событий. 1. Busy Timeout для типов R1b, R5b. 2. Busy Timeout после Write CRC status 3. Write CRC status Timeout 4. Read Data Timeout 0 — Нет ошибки 1 — Timeout	0
3	Command Index Error	RW1C	Флаг выставляется, когда происходит ошибка Command Index во время Command Response. 1 — Ошибка 0 — Нет ошибки	0
2	Command End Bit Error	RW1C	Происходит, когда в конечном бите командного ответа обнаруживается 0. 1 — Ошибка 0 — Нет ошибки	0
1	Command CRC Error	RW1C	Command CRC Error генерируется в 2-х случаях: 1) Если возвращен ответ и флаг Command Time-out Error сброшен в 0, то этот бит устанавливается в 1, когда обнаруживается CRT-ошибка в командном ответе 2) Хост Контроллер обнаруживает конфликт на линии CMD во время выполнения команды. Если Хост Контроллер подает на CMD линию 1, а на следующем такте SDCLK видит 0, то Хост Контроллер должен прервать команду (прекратить управлять линией CMD), и установить этот флаг в 1. Бит Command Timeout Error также должен быть выставлен в 1, чтобы конфликт на линии CMD мог быть распознан. 1 — Сгенерирована CRC ошибка 0 — Нет ошибки	0

Номер бита	Поле	Тип	Описание	Сброс
0	Command Timeout Error	RW1C	Происходит только если не приходит ответ в течении 64 циклов SDCLK после последнего бита команды. Если Хост Контроллер обнаруживает конфликт на линии CMD, то Command CRC Error также должен быть выставлен. Этот бит должен быть выставлен без ожидания 64 циклов SDCLK , потому что команда будет прервана Хост Контроллером. 0 — Нет ошибки 1 — Timeout	0

Таблица 28.26. Соотношения между command CRC error и command time-out error

Command CRC Error	Command Time-out Error	Значение статуса
0	0	Нет ошибок
0	1	Response Timeout Error
1	0	Response CRC Error
1	1	CMD Line Conflict

28.3.19 Регистр Normal interrupt status enable (смещение 034h)

Таблица 28.27. Описание полей регистра Normal interrupt status enable

Номер бита	Поле	Тип	Описание	Сброс
15	Fixed to 0	RO	Хост Контроллер должен контролировать прерывания от ошибок, используя регистр Error Interrupt Status Enable .	0
14	Boot terminate Interrupt enable	RW	0 — Masked 1 — Enabled	0
13	Boot ack rcv enable	RW	0 — Masked 1 — Enabled	0
12	Re-Tuning Event Status Enable	RW	0 — Masked 1 — Enabled	0
11	INT_C Status Enable	RW	Если этот бит установлен в 0, Хост Контроллер должен очистить запрос на прерывание системы. Хост Драйвер может очистить этот бит до обработки INT_C и может установить этот бит снова после того как все запросы на прерывание будут очищены, чтобы избежать ложных срабатываний прерываний.	0
10	INT_B Status Enable	RW	Если этот бит установлен в 0, Хост Контроллер должен очистить запрос на прерывание системы. Хост Драйвер может очистить этот бит до обработки INT_B и может установить этот бит снова после того как все запросы на прерывание будут очищены, чтобы избежать ложных срабатываний прерываний.	0
9	INT_A Status Enable	RW	Если этот бит установлен в 0, Хост Контроллер должен очистить запрос на прерывание системы. Хост Драйвер может очистить этот бит до обработки INT_A и может установить этот бит снова после того как все запросы на прерывание	0

Номер бита	Поле	Тип	Описание	Сброс
			будут очищены, чтобы избежать ложных срабатываний прерываний.	
8	Card Interrupt Status Enable	RW	Если этот бит установлен в 0, Хост Контроллер должен очистить запрос на прерывание системы. Детектирование события Card Interrupt прекращается, когда этот бит сброшен, и возобновляется при его установке. Хост Драйвер может очистить Card Interrupt Status Enable до обработки события Card Interrupt и восстановить его снова после того как все запросы на прерывание от карты будут очищены, чтобы избежать ложных срабатываний прерываний. 0 — Masked 1 — Enabled	0
7	Card Removal Status Enable	RW	0 — Masked 1 — Enabled	0
6	Card Insertion Status Enable	RW	0 — Masked 1 — Enabled	0
5	Buffer Read Ready Status Enable	RW	0 — Masked 1 — Enabled	0
4	Buffer Write Ready Status Enable	RW	0 — Masked 1 — Enabled	0
3	DMA Interrupt Status Enable	RW	0 — Masked 1 — Enabled	0
2	Block Gap Event Status Enable	RW	0 — Masked 1 — Enabled	0
1	Transfer Complete Status Enable	RW	0 — Masked 1 — Enabled	0
0	Command Complete Status Enable	RW	0 — Masked 1 — Enabled	0

28.3.20 Регистр Error interrupt status enable (смещение 036h)

Таблица 28.28. Описание полей регистра Error interrupt status enable

Номер бита	Поле	Тип	Описание	Сброс
15:13	Vendor Specific Error Status Enable	Rsvd	Зарезервировано	0
12	Target Response Error Status Enable	RW	0 — Masked 1 — Enabled	0
11:10	Reserved	Rsvd	Зарезервировано	0
9	ADMA Error Status Enable	RW	0 — Masked 1 — Enabled	0
8	Auto CMD12 Error Status Enable	RW	0 — Masked 1 — Enabled	0
7	Current Limit Error Status Enable	RW	0 — Masked 1 — Enabled	0

Номер бита	Поле	Тип	Описание	Сброс
6	Data End Bit Error Status Enable	RW	0 — Masked 1 — Enabled	0
5	Data CRC Error Status Enable	RW	0 — Masked 1 — Enabled	0
4	Data Timeout Error Status Enable	RW	0 — Masked 1 — Enabled	0
3	Command Index Error Status Enable	RW	0 — Masked 1 — Enabled	0
2	Command End Bit Error Status Enable	RW	0 — Masked 1 — Enabled	0
1	Command CRC Error Status Enable	RW	0 — Masked 1 — Enabled	0
0	Command Time-out Error Status Enable	RW	0 — Masked 1 — Enabled	0

28.3.21 Регистр Normal interrupt signal enable (смещение 038h)

Этот регистр используется для выбора тех сигналов прерываний, которые необходимо включить. Установка одного из этих бит в 1 разрешает генерирование соответствующего прерывания.

Таблица 28.29. Описание полей регистра Normal interrupt signal enable

Номер бита	Поле	Тип	Описание	Сброс
15	Fixed to 0	RO	Хост Драйвер должен контролировать прерывания от ошибок, используя регистр Error Interrupt Signal Enable	0
14	Boot terminate Interrupt signal enable	RW	0 — Masked 1 — Enabled	0
13	Boot ack rcv signal enable	RW	0 — Masked 1 — Enabled	0
12	Re-Tuning Event signal Enable	RW	0 — Masked 1 — Enabled	0
11	INT_C signal Enable	RW	0 — Masked 1 — Enabled	0
10	INT_B signal Enable	RW	0 — Masked 1 — Enabled	0
9	INT_A signal Enable	RW	0 — Masked 1 — Enabled	0
8	Card Interrupt Signal Enable	RW	0 — Masked 1 — Enabled	0
7	Card Removal Signal Enable	RW	0 — Masked 1 — Enabled	0
6	Card Insertion Signal Enable	RW	0 — Masked 1 — Enabled	0
5	Buffer Read Ready Signal Enable	RW	0 — Masked 1 — Enabled	0
4	Buffer Write Ready Signal Enable	RW	0 — Masked 1 — Enabled	0

Номер бита	Поле	Тип	Описание	Сброс
3	DMA Interrupt Signal Enable	RW	0 — Masked 1 — Enabled	0
2	Block Gap Event Signal Enable	RW	0 — Masked 1 — Enabled	0
1	Transfer Complete Signal Enable	RW	0 — Masked 1 — Enabled	0
0	Command Complete Signal Enable	RW	0 — Masked 1 — Enabled	0

28.3.22 Регистр Error interrupt signal enable (смещение 03Ah)

Этот регистр используется для выбора тех сигналов прерываний, которые необходимо включить. Установка одного из этих бит в 1 разрешает генерирование соответствующего прерывания.

Таблица 28.30. Описание полей регистра Error interrupt signal enable

Номер бита	Поле	Тип	Описание	Сброс
15:13	Vendor Specific Error Signal Enable	Rsvd	Зарезервировано	0
12	Target Response Error Signal Enable	RW	0 — Masked 1 — Enabled	0
11:10	Reserved	Rsvd	Зарезервировано	0
9	ADMA Error Signal Enable	RW	0 — Masked 1 — Enabled	0
8	Auto CMD12 Error Signal Enable	RW	0 — Masked 1 — Enabled	0
7	Current Limit Error Signal Enable	RW	0 — Masked 1 — Enabled	0
6	Data End Bit Error Signal Enable	RW	0 — Masked 1 — Enabled	0
5	Data CRC Error Signal Enable	RW	0 — Masked 1 — Enabled	0
4	Data Timeout Error Signal Enable	RW	0 — Masked 1 — Enabled	0
3	Command Index Error Signal Enable	RW	0 — Masked 1 — Enabled	0
2	Command End Bit Error Signal Enable	RW	0 — Masked 1 — Enabled	0
1	Command CRC Error Signal Enable	RW	0 — Masked 1 — Enabled	0
0	Command Time-out Error Signal Enable	RW	0 — Masked 1 — Enabled	0

28.3.23 Регистр Auto CMD error status (смещение 03Ch)

Этот регистр используется для того отображения ошибок при CMD12, ответе во время Auto CMD12, и CMD23, ответе во время Auto CMD23. Из этого регистра Хост Драйвер может

определить какого типа Auto CMD12 / CMD23 произошла ошибка. Ошибки Auto CMD23 отображаются в битах 04-01. Этот регистр действителен только тогда, когда установлен флаг Auto CMD Error.

Таблица 28.31. Описание полей регистра Auto CMD error status

Номер бита	Поле	Тип	Описание	Сброс
15:8	–	Rsvd	Зарезервировано	0
7	Command Not Issued By Auto CMD12 Error	ROC	Этот бит установлен в 0, если Auto CMD Error генерируется командой Auto CMD23 0 — Нет ошибок 1 — Не выполнено	0
6:5	–	Rsvd	Зарезервировано	0
4	Auto CMD Index Error	ROC	Происходит при появлении ошибки связанной с Command Index в ответе на посланную команду 0 — Нет ошибок 1 — Ошибка	0
3	Auto CMD End Bit Error	ROC	Появляется, когда обнаруживается 0 в конечном бите ответной команды 0 — Нет ошибок 1 — Ошибка конечного бита	0
2	Auto CMD CRC Error	ROC	Появляется, когда обнаруживается CRC-ошибка в ответной команде 0 — Нет ошибок 1 — CRC ошибка	0
1	Auto CMD Timeout Error	ROC	Происходит, если ответ не приходит в течении 64 циклов SDCLK после отправки конечного бита команды. Если данный бит установлен, то остальные биты ошибок (D04-D02) не имеют значения. 0 — Нет ошибок 1 — Timeout ошибка	0
0	Auto CMD12 not Executed	ROC	Если передача множественного блока данных не началась вследствие ошибки команды, этот бит не устанавливается, по-скольку в этом нет необходимости для запуска Auto CMD12. Установка этого бита в 1 означает, что Хост Контроллер не может запустить Auto CMD12 , чтобы остановить передачу множественного блока данных вследствие неких ошибок. Если данный бит установлен, то остальные биты ошибок (D04-D02) не имеют значения. Этот бит установлен в 0, если Auto CMD Error генерируется командой Auto CMD23	0

Таблица 28.32. Соотношения между Auto CMD12 CRC error и Auto CMD12 timeout error

Auto Cmd12 CRC Error	Auto CMD12 Timeout Error	Варианты ошибки
0	0	Нет ошибок
0	1	Response Timeout Error
1	0	Response CRC Error
1	1	CMD Line Conflict

28.3.24 Регистр Host control 2 (смещение 03Eh)

Таблица 28.33. Описание полей регистра Host control 2

Номер бита	Поле	Тип	Описание	Сброс
15	Preset Value Enable	RW	1 — Автоматическая выборка параметров из предустановленных значений включена 0 — SDCLK и мощность драйвера контролируются Хост Драйвером Если этот бит сброшен в 0, параметры SDCLK Frequency Select, Clock Generator Select в регистре Clock Control и Driver Strength Select в регистре Host Control 2 устанавливаются Хост Драйвером. Если это бит установлен в 1, SDCLK Frequency Select, Clock Generator Select в регистре Clock Control и Driver Strength Select в регистре Host Control 2 выбираются Хост Контроллером самостоятельно из регистра Preset Value .	0
14	Asynchronous Interrupt Enable	RW	Если карта поддерживает асинхронные прерывания, то этот бит может быть установлен в 1. Параметр Asynchronous Interrupt Support в регистре Capabilities также должен быть установлен в 1. Асинхронные прерывания эффективны, когда в 4-х битном режиме используется прерывание от линии DAT[1] (и Interrupt Pin Select в регистре Shared Bus Control сброшен в 0). Если рассматриваемый бит сброшен в 0, то Хост Драйвер может остановить SDCLK в течении периода асинхронного прерывания для экономии энергии. В течении этого периода Хост Контроллер продолжает передавать Card Interrupt хосту, в случае его генерации картой. 1 — Разрешено 0 — Запрещено	0
13:8	–	Rsvd	Зарезервировано	0
7	Sampling Clock Select	RW	Этот бит устанавливается в процессе процедуры подстройки, когда сброшен флаг Execute Tuning . Запись 1 в данный бит не имеет значения и игнорируется. Установка в 1 означает, что подстройка завершена успешно, а 0 — подстройка не прошла. Хост Контроллер использует данный бит для выбора управляющего клока для управления шинами CMD и DAT. Этот бит сбрасывается записью 0. Изменение этого бита не разрешается пока Хост Контроллер принимает ответ или читает блок данных. 1 — Подстраиваемая частота используется для управления данными 0 — Фиксированная частота используется для управления данными	0
6	Execute Tuning	RWAC	Этот бит устанавливается в 1, чтобы начать процедуру подстройки, и автоматически	0

Номер бита	Поле	Тип	Описание	Сброс
			сбрасывается, когда процедура подстройки Заканчивается. Результат подстройки отображается в параметре Sampling Clock Select. Процедура подстройки прерывается при записи 0 в этот бит. 1 — Выполняется подстройка 0 — Нет подстройки либо подстройка завершена.	
5:4	Driver Strength Select	RW	Эти биты используются выходным драйвером Хост Контроллера при использовании питания 1.8В. В режиме использования 3.3В это поле не используется. Это поле может быть выставлено в зависимости от того, какие значения имеют параметры Driver Type A, C и D в регистре Capabilities . Если Preset Value Enable = 0, то данные биты выставляется Хост Контроллером, Если Preset Value Enable = 1, это поле автоматически заполняется, используя данные регистра Preset Value . 00b выбран Driver Type B (по умолчанию) 01b выбран Driver Type A 10b выбран Driver Type C 11b выбран Driver Type D	0
3	1.8V Signaling Enable	RW	Данный бит управляет регулятором напряжения I/O ячейки. Изменение этого бита из 0 в 1 запускает изменение напряжения сигналов с 3.3В на 1.8В. Внешний регулятор питания должен стабилизироваться в течении 5 мс. Хост Контроллер очищает этот бит, если переключение на 1.8В не удалось. Изменение этого бита из 1 в 0 запускает изменение напряжения сигналов с 1.8В на 3.3В. Внешний регулятор питания должен стабилизироваться в течении 5 мс. Хост Драйвер может установить этот бит в 1, если Хост Контроллер поддерживает управление сигналами в режиме питания 1.8В (один из соответствующих битов SDR50, SDR104 или DDR50 в регистре Capabilities установлен в 1) и если карта или другое подключенное устройство поддерживает UHS-I	0
2:0	UHS Mode Select	RW	Это поле используется для выбоа одного из режимов UHS-I и имеет силу, если параметр 1.8V Signaling Enable равен 1. Если флаг Preset Value Enable регистра Host Control 2 установлен в 1, то Host Controller должен устанавливливать параметры SDCLK Frequency Select, Clock Generator Select в регистре Clock Control и Driver Strength Select в соответствии данными регистра Preset Value . В этом случае одно из предустановленных значений регистра выбирается из данного поля. Хост Драйверу следует сбросить SD Clock Enable	0

Номер бита	Поле	Тип	Описание	Сброс
			до внесения изменений в это поле во избежание сбоев в клоке. После установки этого поля, должен восстановить SD Clock Enable вновь. 000b - SDR12 001b - SDR25 010b - SDR50 011b - SDR104 100b - DDR50 101b - 111 зарезервировано Когда для SDIO-карты выбраны режимы SDR50, SDR104 или DDR50 , детектирование прерываний во время разрывов в блоках не используется. Для этих режимов также изменены временные соотношения для функционала Read Wait . Для получения подробностей обратитесь к SDIO Specification Version 3.00 .	

28.3.25 Регистр Capabilities (смещение 040h)

Хост Драйвер может получить из этого регистра информацию, специфичную для данной реализации Хост Контроллера. Хост Контроллер получает эти данные из конфигурационных регистров (расположенных по адресам 100h...118h) во время инициализации при включении питания ил после общего сброса Хост Контроллера.

Таблица 28.34. Описание полей регистра Capabilities

Номер бита	Поле	Тип	Описание	Сброс
63:58	–	Rsvd	Зарезервировано	0
57	SPI block mode	Hwinit	Режим Spi block 0 — не поддерживается 1 — поддерживается	
56	SPI mode	Hwinit	Режим Spi 0 — не поддерживается 1 — поддерживается	
55:48	Clock Multiplier	Hwinit	В этом поле отображается значение множитель программируемого генератора частоты (подробнее в регистре Clock Control). Значение 00h означает, что программируемый генератор частоты не поддерживается Хост Контроллером. FFh Clock Multiplier M = 256 .... 02h Clock Multiplier M = 3 01h Clock Multiplier M = 2 00h Clock Multiplier is Not Supported	
47:46	Re-tuning modes	Rsvd	Это поле определяет возможности перенастройки Хост Контроллера, а также то, как управлять длиной пересылаемой информации и таймером перенастрйки с помощью Хост Драйвера. 00 — Mode1	

Номер бита	Поле	Тип	Описание	Сброс
			01 — Mode2 10 — Reserved 11 — Reserved Существует 2 типа перенастройки синхронизации: с помощью запроса на перенастройку и с помощью экспирации таймера перенастройки. При получении запроса на перенастройку, Хост Контроллер должен провести соответствующую процедуру только перед выполнением следующей команды.	
45	Use Tuning for SDR50	HwInit	Если данный бит установлен, то Хост Контроллеру будет доступна процедура подстройки в режиме SDR50. (В режиме SDR104 подстройка всегда доступна) 1 — При SDR50 подстройка доступна 0 — При SDR50 подстройка не доступна	
44	–	Rsvd	Зарезервировано	0
43:40	Timer count for Re-Tuning	HwInit	Это поле отображает инициализационное значение таймера для перенастройки 0h — получить информацию из другого источника 1h = 1 секунда 2h = 2 секунды 3h = 4 секунды 4h = 8 секунд .... n = 2^(n-1) секунд .... Bh = 1024 секундного Fh...Ch — Зарезервировано	
39	–	Rsvd	Зарезервировано	0
38	Driver Type D Support	HwInit	Этот бит показывает, поддерживает ли Driver Type D управление сигналами при питании 1.8В 1 — Driver Type D поддерживает 0 — Driver Type D не поддерживает	
37	Driver Type C Support	HwInit	Этот бит показывает, поддерживает ли Driver Type C управление сигналами при питании 1.8В 1 — Driver Type C поддерживает 0 — Driver Type C не поддерживает	
36	Driver Type A Support	HwInit	Этот бит показывает, поддерживает ли Driver Type A управление сигналами при питании 1.8В 1 — Driver Type A поддерживает 0 — Driver Type A не поддерживает	
35	–	Rsvd	Зарезервировано	0
34	DDR50 Support	HwInit	1 — поддерживается режим DDR50 0 — не поддерживается режим DDR50	
33	SDR104 Support	HwInit	1 — поддерживается режим SDR104 0 — не поддерживается режим SDR104	
32	SDR50 Support	HwInit	Если поддерживается SDR104 , то данный бит должен быть установлен в 1. 40-й бит отображает, доступна ли для режима SDR50 подстройка или нет 1 — поддерживается режим SDR50	

Номер бита	Поле	Тип	Описание	Сброс
			0 — не поддерживается режим SDR50	
31:30	Slot Type	HwInit	Это поле показывает, какой тип слота используется контроллером. 00b — слот с извлекаемой картой (по умолчанию) 01b — встроенный слот для единичного устройства. Это означает, что только одно неизвлекаемое устройство подключено к шине SD. 10b — слот с совместно используемой шиной. Данный слот может быть установлен, если Хост Контроллер поддерживает регистр Shared Bus Control . 11b — Зарезервировано Стандартный Хост Драйвер контролирует только извлекаемый тип карты или одно встроенное устройство, подключенное к слоту шины SD. Если слот будет сконфигурирован для совместно используемой шины, Стандартный Хост Контроллер не сможет контролировать встроенное устройство, подключенное к общей шине. Слот для общей шины должен контролироваться специально разработанным хост драйвером.	
29	Asynchronous Interrupt Support	HwInit	За подробностями по работе асинхронных прерываний обратитесь к *SDIO Specification Version 3.00* . 1 — Асинхронные прерывания поддерживаются 0 — Асинхронные прерывания не поддерживаются	
28	64-bit System Bus Support	HwInit	1 — поддерживается 64-х битный системный адрес 0 — не поддерживается	
27	–	Rsvd	Зарезервировано	0
26	Voltage Support 1.8 V	HwInit	1 — Поддерживается 1.8V 0 — Не поддерживается 1.8V	
25	Voltage Support 3.0 V	HwInit	1 — Поддерживается 3.0V 0 — Не поддерживается 3.0V	
24	Voltage Support 3.3 V	HwInit	1 — Поддерживается 3.3V 0 — Не поддерживается 3.3V	
23	Suspend / Resume Support	HwInit	Этот бит отображает, поддерживается ли функционал Suspend / Resume . Если этот бит сброшен в 0, то процедуры Suspend и Resume не поддкрживаются, Хост Драйвер не должен запускать ни одну из этих команд. 1 — Поддерживается 0 — Не поддерживается	
22	SDMA Support	HwInit	Этот бит показывает, поддерживается ли Хост Контроллером использование DMA транзакций напрямую между системной памятью и Хост Контроллером 1 — SDMA поддерживается 0 — SDMA не поддерживается	

Номер бита	Поле	Тип	Описание	Сброс
21	High Speed Support	HwInit	Этот бит показывает, есть ли у Хост Контроллера и у Хост Системы поддержка высокоскоростных режимов и могут ли они поддерживать частоты от 25Mhz до 50 Mhz (для SD)/ от 20MHz до 52MHz (для MMC) 1 — High Speed поддерживается 0 — High Speed не поддерживается	
20	–	Rsvd	Зарезервировано	0
19	ADMA2 Support	HwInit	1 — ADMA2 поддерживается 0 — ADMA2 не поддерживается	
18	Extended Media Bus Support	HwInit	Данный бит показывает способен ли Хост Контолроллер использовать режим 8-и битной шины. Этот бит не имеет силы, если параметр Slot Type установлен в 10b. 1 — Extended Media Bus поддерживается 0 — Extended Media Bus не поддерживается	
17:16	Max Block Length	HwInit	Это значение отображает максимальный размер блока данных, который Хост Драйвер может читать и писать в буфер Хост Контроллера. Буфер должен передавать блоки с таким размером без цикла ожидания. Может быть определено 3 возможных размера блока: 00 - 512 byte 01 - 1024 byte 10 - 2048 byte (по умолчанию) 11 - Reserved	
15:8	Base Clock Frequency for SD Clock	HwInit	Исходная частота для SD клока (1) 6-bit Base Clock Frequency Этот режим поддерживается Хост Контроллером ver 1.0 и 2.0. Старшие 2 бита не используются и равны 0. Поддерживается диапазон частот от 10MHz до 63MHz. 11xx xxxxb — не поддерживается 0011 1111b — 63MHz 0000 0011b — 3MHz 0000 0010b — 2MHz 0000 0001b — 1MHz 0000 0000b — Получить информацию другим способом (2) 8-bit Base Clock Frequency Этот режим поддерживается Хост Контроллером ver 3.0. Поддерживается диапазон частот от 10MHz до 255MHz. FFh — 255MHz 03h — 3MHz 02h — 2MHz 01h — 1MHz 00h — Получить информацию другим способом Если реальная частота — 16.5MHz, то значение частоты, которое должно быть выставлено в этом регистре следует округлить в большую сторону. Т.е. Необходимо выставить 00010001b (17MHz), потому что Хост Драйвер использует	

Номер бита	Поле	Тип	Описание	Сброс
			это значение для вычисления значения предделителя частоты (подробнее см. параметр SDCLK Frequency Select в регистре Clock Control) и оно не должно превышать верхнего значения частоты SD. Если в этом поле записан 0, то Хост Ситеме для получения информации нужно воспользоваться другими методами.	
7	Timeout Clock Unit	HwInit	Этот бит показывает, в каких единицах измерения задается частота, используемая для определения Data Timeout Error. 0 - Khz 1 - Mhz	
6	–	Rsvd	Зарезервировано	0
5:0	Timeout Clock Frequency	HwInit	В этом поле отображается значение основной чатоты, используемой для определения Data Time- out Error. Не 0 : от 1Khz до 63Khz либо от 1Mhz до 63Mhz (в зависимости от значения параметра Timeout Clock Unit) 0 : Получить информацию другим способом	

Re-tuning mode

Хост Контроллер не имеет никакой встроенной логики для определения, когда нужно будет выполнить процедуру подстройки. В этом случае Хост Драйвер должен поддерживать все типы перенастройки, используя Re-Tuning Timer. Для разрешения вставки процедуры подстройки в течении пересылки данных, длина данных на одну команду чтения/записи не должна превышать 4 MB.

Таблица 28.35. Режимы подстройки

Bit47-46	Re-Tuning Mode	Data length	Timer Modes
00b	Mode1	4MB (Max.)	Всегда включен
01b	Mode1	4MB (Max.)	Останавливается в течение пересылки данных
10b	Reserved	Reserved	Reserved
11b	Reserved	Reserved	Reserved

28.3.26 Регистр Maximum current capabilities (смещение 048h)

Таблица 28.36. Описание полей регистра Maximum current capabilities

Номер бита	Поле	Тип	Описание	Сброс
63:56	–	Rsvd	Зарезервировано	0
55:48	–	Rsvd	Зарезервировано	0
47:40	–	Rsvd	Зарезервировано	0
39:32	–	Rsvd	Зарезервировано	0
31:24	–	Rsvd	Зарезервировано	0

Номер бита	Поле	Тип	Описание	Сброс
23:16	Maximum Current for 1.8V	HwInit	Максимальный ток для 1.8V	
15:8	Maximum Current for 3.0V	HwInit	Максимальный ток для 3.0V	
7:0	Maximum Current for 3.3V	HwInit	Максимальный ток для 3.3V	

Значения токов, задаваемые в этом регистре, определены в следующей таблице

Таблица 28.37. Сила тока, соответствующая значениям регистра

Значение в регистре	Величина силы тока
0	Получить информацию другим способом
1	4mA
2	8mA
3	12mA
...	
255	1020mA

28.3.27 Регистр Force event для Auto CMD error status (смещение 050h)

Регистр Force Event не является физически реализованным регистром. Скорее это адрес, по которому может происходить запись в регистр Auto CMD12 Error Status. Запись 1 в один из битов данного регистра, установит соответствующий бит в регистре Auto CMD12 Error Status . Запись 0 игнорируется.

Таблица 28.38. Описание полей регистра Force event для Auto CMD error status

Номер бита	Поле	Тип	Описание	Сброс
15:8	–	Rsvd	Зарезервировано	0
7	Force Event for command not issued by Auto CMD12 Error	RW	1 — прерывание сгенерировано 0 — нет прерываний	0
6:5	–	Rsvd	Зарезервировано	0
4	Force Event for Auto CMD Index Error	RW	1 — прерывание сгенерировано 0 — нет прерываний	0
3	Force Event for Auto CMD End bit Error	RW	1 — прерывание сгенерировано 0 — нет прерываний	0
2	Force Event for Auto CMD CRC Error	RW	1 — прерывание сгенерировано 0 — нет прерываний	0
1	Force Event for Auto CMD timeout Error	RW	1 — прерывание сгенерировано 0 — нет прерываний	0
0	Force Event for Auto CMD12 NOT Executed	RW	1 — прерывание сгенерировано 0 — нет прерываний	0

28.3.28 Регистр Force event для error interrupt status (смещение 052h)

Регистр Force Event не является физически реализованным регистром. Скорее это адрес, по которому может происходить запись в регистр Error Interrupt Status. Запись по этому адресу

будет отображаться в регистр Error Interrupt Status, если установлены соответствующие разрешающие биты в регистре Error Interrupt Status Enable. Запись 0 игнорируется.

Таблица 28.39. Описание полей регистра Force event для error interrupt status

Номер бита	Поле	Тип	Описание	Сброс
15:13	Force Event for Vendor Specific Error Status	Rsvd	Зарезервировано	0
12	Force event for Target Response error	RW	1 — прерывание сгенерировано 0 — нет прерываний	0
11:10	–	Rsvd	Зарезервировано	0
9	Force Event for ADMA Error	RW	1 — прерывание сгенерировано 0 — нет прерываний	0
8	Force Event for Auto CMD Error	RW	1 — прерывание сгенерировано 0 — нет прерываний	0
7	Force Event for Current Limit Error	RW	1 — прерывание сгенерировано 0 — нет прерываний	0
6	Force Event for Data End Bit Error	RW	1 — прерывание сгенерировано 0 — нет прерываний	0
5	Force Event for Data CRC Error	RW	1 — прерывание сгенерировано 0 — нет прерываний	0
4	Force Event for Data Timeout Error	RW	1 — прерывание сгенерировано 0 — нет прерываний	0
3	Force Event for Command Index Error	RW	1 — прерывание сгенерировано 0 — нет прерываний	0
2	Force Event for Command End Bit Error	RW	1 — прерывание сгенерировано 0 — нет прерываний	0
1	Force Event for Command CRC Error	RW	1 — прерывание сгенерировано 0 — нет прерываний	0
0	Force Event for Command Time-out Error	RW	1 — прерывание сгенерировано 0 — нет прерываний	0

28.3.29 Регистр ADMA error status (смещение 054h)

Когда происходит событие ADMA Error Interrupt, поле ADMA Error States в данном регистре сохраняет состояние ADMA, а регистр ADMA System Address сохраняет адрес, ошибки дескриптора.

Таблица 28.40. Описание полей регистра ADMA error status

Номер бита	Поле	Тип	Описание	Сброс
7:3	–	Rsvd	Зарезервировано	0
2	ADMA Length Mismatch Error	RO	Эта ошибка происходит в двух следующих случаях: В то время, как устанавливается значение While Block Count Enable , общая длина данных, определяемая таблицей дескриптора может отличаться от той, что задается параметрами	0

Номер бита	Поле	Тип	Описание	Сброс
			Block Count и Block Length. Общая длина данных может не уместиться в длину блока. 1 — Ошибка 0 — Нет ошибки	
1:0	ADMA Error State	RO	Это поле показывает состояние ADMA при появлении ошибки в течение передачи данных по ADMA . Это поле никогда не становится равным 10b, потому что ADMA никогда не останавливается в этом состоянии. D01 – D00 : ADMA Error State во время появления ошибки. Содержание регистра SYS_SDR 00 - ST_STOP (Stop DMA) Указывает на следующий после ошибки дескриптор 01 - ST_FDS (Fetch Descriptor) Указывает на ошибочный дескриптор 10 - состояние не используется и никогда не выставляется 11 - ST_TFR (Transfer Data) Указывает на следующий после ошибки дескриптор	0

28.3.30 Регистр ADMA system address (смещение 058h)

Таблица 28.41. Описание полей регистра ADMA system address

Номер бита	Поле	Тип	Описание	Сброс
63:0	ADMA System Address	RW	Этот регистр сохраняет адрес байта выполняемой команды дескрипторной таблицы. 32-х битный адрес дескриптора использует младшие 32 бита регистра. При старте ADMA Хост Драйвер должен установить начальный адрес дескрипторной таблицы. При каждой выборке строки дескриптора ADMA инкрементирует этот адрес, который указывает на следующую строку. При появлении ADMA Error Interrupt этот регистр должен сохранить правильное значение дескрипторного адреса, относящегося к сотоянию ADMA. Хост Драйвер должен программировать таблицу дескрипторов с границей в 32 бита и установить 32-битный граничный адрес в этот регистр. ADMA игнорирует младшие 2 бита этого регистра и принимает их равными 00b. 32-bit Address ADMA Register Value 32- bit System Address xxxxxxxx 00000000h 00000000h xxxxxxxx 00000004h 00000004h .... xxxxxxxx FFFFFFFCh FFFFFFFCh	0

28.3.31 Регистры Preset Values (смещения 06Fh — 060h)

Таблица 28.42. Общее описание регистров Preset Values

Смещение	Регистры Preset Value	Напряжения сигналов
060h	Preset Value for Initialization	3.3V или 1.8V
062h	Preset Value for Default Speed	3.3V
064h	Preset Value for High Speed	3.3V
066h	Preset Value for SDR12	1.8V
068h	Preset Value for SDR25	1.8V
06Ah	Preset Value for SDR50	1.8V
06Ch	Preset Value for SDR104	1.8V
06Eh	Preset Value for DDR50	1.8V

На основе выбранного скоростного режима шины входит в силу значения одного из регистров Preset Value (06Eh — 062h). Когда параметр Preset Value Enable в регистре Host Control 2 установлен в 1, то SDCLK Frequency Select и Clock Generator Select в регистре Clock Control, а также Driver Strength Select в регистре Host Control 2 устанавливаются автоматически на основе выбранного скоростного режима шина. Это означает, что Хост Драйверу нет необходимости устанавливать эти значения, если разрешено использование предустановленных значений. Значения из Preset Value for Initialization (060h) не выбираются с помощью скоростного режима шины. До начала инициализационной последовательности Хост Драйвер должен задать значение частоты для проведения инициализации в SDCLK Frequency Select регистра Clock Control. Preset Value Enable может быть выставлен после завершения инициализации.

Каждый из регистров Preset Values имеет следующую структуру:

Таблица 28.43. Описание полей какого-либо регистра из множества Preset Values

Номер бита	Поле	Тип	Описание	Сброс
15:14	Driver Strength Select Value	Hwinit	Driver Strength поддерживается скоростными режимами, использующими сигналы 1.8V. Это поле не имеет значения для режимов с питанием сигналов от 3.3V. 11b — выбран Driver Type D 10b — выбран Driver Type C 01b — выбран Driver Type A 00b — выбран Driver Type B	
13:11	–	Rsvd	Зарезервировано	0
10	Clock Generator Select Value	Hwinit	Этот бит имеет силу, когда Хост Контроллер поддерживает программируемый тактовый генератор. 1 — Программируемый тактовый генератор. 0 — Хост контроллер ver 2.0, совмещенный тактовый генератор.	
9:0	SDCLK Frequency Select Value	Hwinit	Предустановленное 10-битное значение, задающее SDCLK Frequency Select в регистре Clock Control.	

28.3.32 Регистр Boot Timeout control (смещение 70h)

Таблица 28.44. Описание полей регистра Boot Timeout control

Номер бита	Поле	Тип	Описание	Сброс
31:3	Boot Data Time-out Counter Value	RW	Это значение определяет интервал через который детектируются time-outs на линии DAT в процессе процедуры загрузки eMMC карты. Это значение — количество тактов SD клока.	0

28.3.33 Регистр Slot interrupt status (смещение 0FCh)

Таблица 28.45. Описание полей регистра Slot interrupt status

Номер бита	Поле	Тип	Описание	Сброс
15:8	–	Rsvd	Зарезервировано	0
7:1	–	Rsvd	Зарезервировано	0
0	Interrupt Signal for Slot#0	ROC	Этот статусный бит является результатом операции OR всех включенных нормальных прерываний и включенных прерываний от ошибок.	0

28.3.34 Регистр Host controller version (смещение 0FEh)

Таблица 28.46. Описание полей регистра Host controller version

Номер бита	Поле	Тип	Описание	Сброс
15:8	Vendor Version Number	Rsvd	Vendor Version Number установлен в 0x10 (1.0)	10h
7:0	Specification Version Number	Rsvd	Host Controller Version Number установлен в 0x02 (SD Host Specification Version 3.00)	02h

28.4 Примеры настройки частот и режимов

Таблица 28.47. Предустановленные значения для различных конфигурационных режимов при различных частотах

Регистр	*Значение частоты SDCLK [9:0], FBASE				
	При основной частоте 50МГц	При основной частоте 100МГц	При основной частоте 200МГц	При основной частоте 250МГц	
				Значение регистра, N	Частота передачи
corecfg_initpresetval[12:0]	0x40	0x80	0x100	0x140	390 кГц
corecfg_dsppresetval[12:0]	0x01	0x02	0x04	0x05	25 МГц
corecfg_hsppresetval[12:0]	0x00	0x01	0x02	0x03	41.7 МГц
corecfg_sdr12presetval[12:0]	0x01	0x02	0x04	0x05	25 МГц
corecfg_sdr25presetval[12:0]	0x00	0x01	0x02	0x03	41.7 МГц

Регистр	*Значение частоты SDCLK [9:0], F_{BASE}				
	При основной частоте 50МГц	При основной частоте 100МГц	При основной частоте 200МГц	При основной частоте 250МГц	
				Значение регистра, N	Частота передачи
corecfg_sdr50presetval[12:0]	NA	0x00	0x01	0x02	62.5 МГц
corecfg_sdr104presetval[12:0]	NA	NA	0x00	0x01	125 МГц
corecfg_ddr50presetval[12:0]	0x00	0x01	0x02	0x03	41.7 МГц

***Примечание**. Значение частоты SDCLK рассчитывается по следующей формуле:

$SDCLK=F_{BASE} / 2*N,$

Таблица 28.48. Максимальные значения частот для различных режимов работы SD хост-контроллера

Режим работы	Максимальная скорость шины (МБ/с)	Максимальная тактовая частота (частота тактирования SD-карты)(МГц)	Напряжение сигналов (В)
SDR104	104	208	1.8
SDR50	50	100	1.8
DDR50	50	50	1.8
SDR25	25	50	1.8
SDR12	12.5	25	1.8
High Speed	25	50	3.3
Default Speed	12.5	25	3.3

29. КОНТРОЛЛЕР NAND FLASH (NAND)

Контроллер предназначен для подключения NAND флэш-памяти и обеспечивает работу в соответствии со стандартами ONFI 1.0, 2.0, 2.1, 2.2, 2.3, 3.0, 3.1, 3.2.

Основные параметры:

- Поддержка флэш-памяти с ёмкостью до 1024 Гбит.
- Поддержка памятей производства Micron, Samsung, ST-Micro и др.
- Поддержка всех обязательных команд и поддержка отдельных необязательных
- Полный доступ к spare-области
- Поддержка скоростей до 300 MT/s (макс. режим - NV-DDR2 mode 6 до 150 МГц).
- Поддержка Interleaving-команд:
    - Page Program Interleaving
    - Copyback Program Interleaving
    - Block Erase Interleaving
    - Read Interleaving
    - Cache Interleaving
- Поддержка Multi LUN/DIE операций
- Поддержка Small Data Move
- Поддержка Change Row Address
- Поддержка Reset LUN
- Поддержка EZ-NAND устройств.
- Поддержка механизма Chip_en Pin reduction
- Поддержка ODT Configure
- Поддержка On-die Termination

Особенности реализации:

- Размер страницы – 512, 2K, 4K, 8K, 16K байт
- Программируемый тайминг
- Адресные циклы – 4, 5
- Включение/отключение ECC
- Ширина данных флэш
    - 8-бит для SDR, NV-DDR и NV-DDR2
    - 16-бит только для SDR
- Размер внутренней статической памяти – 4 Кбайта
- **Не поддерживается** параллельное подключение двух 8-ми битных флэш

Коррекция и обнаружение ошибок:

- Обнаружение и коррекция ошибок в режиме SLC (код Хэмминга)
    - Исправление 1-битной ошибки на каждые 512 байт
    - Обнаружение 2-битных ошибок на каждые 512 байт
- Работа в следующих режимах с коррекцией в кодах БЧХ:
    - MLC-4. Коррекция и обнаружение не более 4 ошибок на каждые 512 байт
    - MLC-8. Коррекция и обнаружение не более 8 ошибок на каждые 512 байт
    - MLC-12. Коррекция и обнаружение не более 12 ошибок на каждые 512 байт
    - MLC-16. Коррекция и обнаружение не более 16 ошибок на каждые 512 байт
    - MLC-24. Коррекция и обнаружение не более 24 ошибок на каждые 1024 байта

Интерфейс данных флэш:

- ONFI 1.0, 2.0, 2.1, 2.2, 2.3, 3.0, 3.1 и 3.2
- Поддержка SDR режимов 0-5
- Поддержка NV-DDR режимов 0-5

- Поддержка NV-DDR2 режимов 0-6
- 8-битная шина данных для SDR, NV-DDR и NV-DDR2
- 16-битная шина данных только для режима SDR

Пересылки данных в режимах:

- PIO. Полностью управляется CPU
- Master DMA. Инициатором является контроллер, минимальное участие CPU

29.1 Общая информация

Структурная схема NAND:

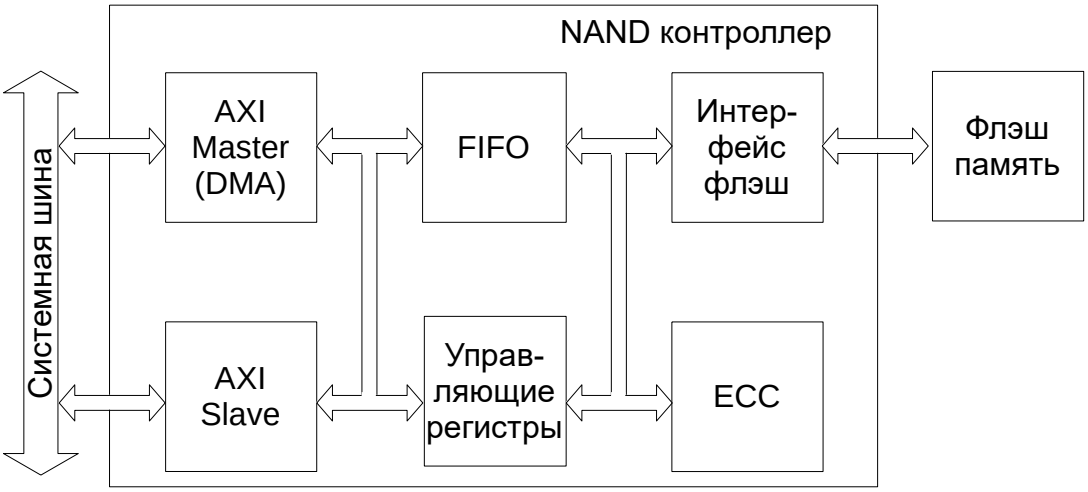


Рисунок 29.1. NAND. Структурная схема

Интерфейс флэш. Интерфейс флэш устанавливает сигналы управления памяти и обеспечивает обмен в соответствии со стандартами ONFI.

AXI Slave. Через ведомый порт осуществляется доступ процессора к управляющим регистрам NAND, операции чтения/записи флэш-памяти.

AXI Master. NAND может работать как ведущее устройство на системной шине в режиме MDMA (Master DMA). При операции записи в память мастер считывает данные из системной памяти и сохраняет их в FIFO. При операции чтения из флэш-памяти мастер считывает данные из FIFO и сохраняет в системную память.

FIFO. При операции чтения данные, полученные из флэш-памяти, помещаются в FIFO. А затем считываются из FIFO либо процессором (через регистры), либо встроенным блоком DMA. При операции записи во флэш-память данные предварительно загружаются в FIFO

либо процессором, либо блоком DMA. Затем данные из FIFO записываются во флэш-память.

ECC (Error-Correcting Code). Блок ECC обеспечивает обнаружение и коррекцию ошибок в режиме HAM (код Хэмминга) и в режиме BCH (код Боуза-Чоудхури/BCH/БЧХ).

Регистры. Процессор управляет конфигурацией и работой NAND через регистры. Конфигурирование включает в себя установку временных параметров, настройку адреса, размера пересылки и т.д. Через регистры производится запуск выполнения команд, чтение/запись данных из FIFO.

29.1.1 Определения и термины

Страница (Page) – наименьшая адресуемая единица для операций чтения и программирования. Страница состоит из множества байтов или слов. (см. ONFI)

Страница параметров (Parameter Page). Структура данных во флэш-памяти, которая описывает организацию флэш-памяти, поддерживаемые свойства, временные режимы и другие поведенческие параметры. Страница параметров считывается из флэш-памяти с помощью команды Read Parameter Page. (см. ONFI)

Блок (Block). Блок состоит из множества страниц и является наименьшей адресуемой единицей для операции Erase. (см. ONFI)

Регистр страницы (Page Register). Регистр страницы используется для временного хранения данных до того, как они будут записаны в страницу в массиве флэш-памяти или после их чтения из массива флэш-памяти. (см. ONFI)

LUN – наименьший элемент, который может независимо выполнять команды и сообщать о своём состоянии. LUN содержит как минимум один регистр страницы и массив флэш-памяти. (см. ONFI)

Целевое устройство (Target) – независимый компонент флэш-памяти с собственным сигналом CE#. Целевое устройство может состоять из нескольких LUN. (см. ONFI)

Массив флэш (Flash Array). Массив запоминающих ячеек флэш-памяти.

Адрес флэш. Адрес состоит из адреса строки (Row Address) и адреса столбца (Column Address). (см. ONFI)

Адрес строки (Row Address) – адрес LUN, блока и страницы к которой производится обращение. (см. ONFI)

Адрес столбца (Column Address). Адрес столбца определяет байт или слово внутри страницы, к которой производится обращение. (см. ONFI)

Interleaved/Multiplane операции. При interleaved/multi-plane операциях множество команд одного типа может одновременно выполнятся над различными блоками одного LUN. (см. ONFI)

NAND поддерживает следующие multi-plane операции:

1. Команда Page Program.
2. Команда Copyback Program.
3. Команда Block Erase.
4. Команда Read.
5. Команда Cache Page Program.

Пакет. Пакет - порция данных, которую считывает или записывает NAND во флэш-память за одну пересылку. Количество байт внутри пакета и общее количество пакетов настраивается программно.

Spare байты/область памяти. Дополнительная резервная область страницы. В данной области размещаются контрольные данные, позволяющие обнаруживать и корректировать ошибки в основной области.

29.2 Регистры NAND

При чтении и записи регистров могут быть использованы данные размером в 8, 16 и 32 разряда.

29.2.1 Карта памяти NAND

Таблица 29.1. Список регистров NAND и их адреса

Сокращённое название	Смещение	Тип	Название
PACKET	0x0	RW	Регистр настройки пакетов
MEMADDR1	0x4	RW	Регистр адреса 1
MEMADDR2	0x8	RW	Регистр адреса 2
COMMAND	0xC	RW	Регистр команды
PROGRAM	0x10	RW	Регистр программы
INTERRUPT_STATUS_EN	0x14	RW	Регистр разрешения статуса прерываний
INTERRUPT_SIGNAL_EN	0x18	RW	Регистр разрешения сигнала прерываний
INTERRUPT_STATUS	0x1C	RW	Регистр статуса прерываний
READY_BUSY	0x20	RO	Регистр готовности-занятости
DMA_SYS_ADDR1	0x24	RW	Системный адрес DMA 1
FLASH_STATUS	0x28	RO	Регистр статуса флэш-памяти
TIMING	0x2C	RW	Регистр временных параметров
BUFFER_DATA	0x30	RW	Регистр данных буфера
ECC	0x34	RW	Регистр ECC
ECC_ERROR_CNT	0x38	RO	Счётчик ошибок ECC
ECC_SPARE_CMD	0x3c	RW	Регистр команд ECC
ERROR_CNT_1BIT	0x40	RW	Счётчик 1-битных ошибок
ERROR_CNT_2BIT	0x44	RW	Счётчик 2-битных ошибок
ERROR_CNT_3BIT	0x48	RW	Счётчик 3-битных ошибок
ERROR_CNT_4BIT	0x4C	RW	Счётчик 4-битных ошибок

DMA_SYS_ADDR0	0x50	RW	Системный адрес DMA 0
DMA_BUFFER_BOUNDARY	0x54	RW	Граница DMA пересылки
CPU_RELEASE	0x58	RW	Регистр состояния CPU
ERROR_COUNT_5BIT	0x5c	RW	Счётчик 5-битных ошибок
ERROR_COUNT_6BIT	0x60	RW	Счётчик 6-битных ошибок
ERROR_COUNT_7BIT	0x64	RW	Счётчик 7-битных ошибок
ERROR_COUNT_8BIT	0x68	RW	Счётчик 8-битных ошибок
DATA_INTF	0x6C	RW	Регистр интерфейса данных
–	0x70 - 0xFF	-	Зарезервировано

29.2.2 Регистр PACKET (0x00)

Таблица 29.2. PACKET - регистр настройки пакетов

Поле	Биты	Значение по сбросу	Описание
packet_size	10:0	0	Размер пакета. Количество байт, которое должен обработать контроллер за один пакет. 11'h001 – 1 байт. 11'h002 – 2 байта. ... Типичный размер пакета в режиме коррекции ошибок BCH 4, 8, 12 и 16 бит − 512 байт. Типичный размер пакета в режиме коррекции ошибок BCH 24 бит - 1024 байта.
–	11	0	Зарезервировано.
packet_count	23:12	0	Количество пакетов. 12'h001 – количество пакетов = 1. ... 12'h800 - количество пакетов =2048.
reserved	31:24	0	Зарезервировано.

В следующей таблице показаны особенности настройки регистра PACKET в зависимости от выполняемой команды и выбранного временного режима.

Таблица 29.3. Пример настройки регистра PACKET

Режим	Команды	packet_size	packet_count
SDR	Set Features, Get Features, Read ID(ONFI)	4	1
SDR	Read ID (JEDEC)	5	1
SDR	Read Status, Read Status Enhanced	1	1
NV-DDR/NV-DDR2	Set Features, Get Features, Read ID(ONFI)	8	1
NV-DDR/NV-DDR2	Read ID (JEDEC)	10	1
NV-DDR/NV-DDR2	Read Status, Read Status Enhanced	2	1

Для команд Set Features, Get Features, Read Status, Read Status Enhanced, Read ID в DDR режимах байты данных повторяются дважды.

При последовательном способе работы с флэш-памятью общее число пересылок (packet_size x packet_count) не должно превышать размер страницы флэш-памяти. Для параллельного режима (одновременно с двумя флэш-памятями) максимальный размер

пересылок может быть в два раза больше размера страницы, так как поле packet_count для параллельного режима должно быть в два раза больше, чем для последовательного.

Таблица 29.4. Пример настроек регистра PACKET для команд Read и Program

Параметр/Размер Страницы	512	2k	4k	8k
packet_size xpacket_count (последов.)	512x1	512x 4	512x 8	512x 16
packet_size xpacket_count (парал.)	512x 2	512x 8	512x 16	512x 32
packet_size xpacket_count (последов.)	-	1024x 2	1024x 4	1024x 8
packet_size xpacket_count (парал.)	-	1024x 4	1024x 8	1024x 16

29.2.3 Регистр MEMADDR1 (0x04)

Таблица 29.5. MEMADDR1 - регистр адреса 1

Поле	Биты	Значение по сбросу	Описание
lsb_addr	31:0	0	Адрес памяти (младшие разряды). Например, для флэш 4k, 4plane, 2lun: 12:0 адрес столбца. 15:13 должны быть в нуле 22:16 адрес страницы. Разряды [24:23] – имеют особое значение для interleaved/mulit-plane операций. 2'b00 – адрес/плоскость 0. 2'b01 – адрес/плоскость 1. 2'b10 – адрес/плоскость 2. 2'b11 – адрес/плоскость 3. 31:25 - адрес блока (младшие разряды).

Примечание. Для команд Block Erase и Read Status Enhanced регистр MEMADDR1 должен быть запрограммирован следующим образом:

Разряды [6:0] – адрес страницы.

Разряды [8:7] – выбор одного из чередующихся адресов:

2'b00 – адрес/плоскость 0.

2'b01 – адрес/плоскость 1.

2'b10 – адрес/плоскость 2.

2'b11 – адрес/плоскость 3.

Разряды [17:9] – адрес блока.

Разряды [18] – выбор логического устройства (LUN).

1'b0 - выбран LUN0.

1'b1 - выбран LUN1.

Разряды [31:19] – должны быть равны нулю.

29.2.4 Регистр MEMADDR2 (0x08)

Таблица 29.6. MEMADDR2 - регистр адреса 2

Поле	Биты	Значение по сбросу	Описание
msb_addr	1:0	0	Адрес памяти (старшие разряды). Например, для флэш 4k, 4plane, 2lun: 1:0 – старшие разряды блока 2 – выбор LUN 7-3 в нуле
–	23:8	0	Зарезервировано
bus_width	24	0	Ширина шины данных флэш. 1'b0 – 8 бит 1'b1 – 16 бит
nfc_bch_mode	27:25	0	Режим работы BCH: 3'b000 – 16-битный код ECC 3'b001 – 12-битный код ECC 3'b010 – 8-битный код ECC 3'b011 – 4-битный код ECC 3'b100 – 24-битный код ECC
Mode	29:28	0	Режим работы с флэш-памятями. 2'b00 - Последовательный режим. 2'b01 - Зарезервировано. 2'b10 - Зарезервировано. 2'b11 - Зарезервировано.
chip_select	31:30	0	Выбор флэш-памяти. 2'b00 - флэш-память 0. 2'b01 - флэш-память 1. 2'b10 - зарезервировано 2'b11 - зарезервировано

Примечание. Для команд Block Erase и Read Status Enhanced биты [7:0] регистра MEMADDR2 должны быть равны нулю.

29.2.5 Регистр COMMAND (0x0C)

Таблица 29.7. COMMAND — регистр команды

Поле	Биты	Значение по сбросу	Описание
command1	7:0	8'h0	Код операции первой команды для флэш-памяти.
command2	15:8	8'h0	Код операции второй команды для флэш-памяти.
–	21:16	3'd0	Зарезервировано.
bch_residue_xor	22	1'd0	Только для включенного режима BCH. 1'd0 – Не допускается использовать команду Read сразу после команды Erase. 1'd1 – Допускается использовать команду Read сразу после команды Erase.
page_size	25:23	3'd2	Размер страницы. 3'd0 – 512 байт. 3'd1 – 2K. 3'd2 – 4K. 3'd3 - 8K.

Поле	Биты	Значение по сбросу	Описание
			3’d4 - 16K. 3’d5 - 1K для 16-битных флэш. 6-7 - Зарезервировано.
dma_enable	27:26	2’d0	Режим обмена данными. 2’d0 - Режим PIO. 2’d1 - Зарезервировано 2’d2 - Режим MDMA. 2’d3 - Зарезервировано.
num_addr_cycles	30:28	3’d0	Число циклов адреса в команде для флэш-памяти. 3’d0 – Не используется. 3’d1 – Один цикл адреса. 3’d2- Два цикла адреса. .... 3’d7 – Семь циклов адреса.
ecc_on_off	31	0	Включение/отключение коррекции ошибок. 1 – ECC включён. 0 - ECC выключен.

Примечание. Значение поля **num_addr_cycles** должно быть равно:

- 0 для команд reset, synchronous reset, read_status (см. регистр PROGRAM);
- 1 для команд read_parameter, get_feature, set_feature, read_id (см. регистр PROGRAM);
- 3 для команд erase, read_status_enhanced, reset_lun (см. регистр PROGRAM).
- Определяется значением байта «101» страницы параметров для команд read, write. Например, 101-й байт содержит значения column address cycle [7:4] = 2 и row address cycle [3:0] = 3, в этом случае процессор должен записать в поле num_addr_cycles значение 5 (column address cycle + row address cycle).

Примечание. Для команд, требующих пересылок данных (Read, Program Page и др.), существует несколько способов обмена данными:

В режиме PIO (Processor IO) пересылки данных выполняются процессором.

В режиме MDMA (Master DMA) пересылки данных (см. разряды read, page_program, read_interleaved, change_read_column_enhanced, read_cache_start, read_cache_sequential, read_cache_random, read_cache_end регистра PROGRAM) выполняются контроллером, при минимальном участии процессора.

29.2.6 Регистр PROGRAM (0x10)

Через регистр PROGRAM производится инициирование выдачи команд для флэш-памяти контроллером NAND. Запись единицы в необходимый разряд вызовет выполнение соответствующей команды. После завершения отработки команды NAND сбросит регистр PROGRAM и установит прерывание “Transfer Complete”.

Таблица 29.8. PROGRAM - регистр программы

Поле	Биты	Значение по сбросу	Описание
read	0	1'b0	Операция чтения из флэш-памяти.
multi_die	1	1'b0	Признак Interleaved/Multiplane операции (Page Program, Read, Block Erase).
block_erase	2	1'b0	Операция Block Erase.
read_status	3	1'b0	Операция Read Status.
page_program	4	1'b0	Операция Page Program.
multi_die_rd	5	1'b0	Операция чтения из разных LUN.
read_id	6	1'b0	Операция Read ID.
read_parameter_page	7	1'b0	Операция Read Parameter Page.
reset	8	1'b0	Операция Reset.
get_features	9	1'b0	Операция Get Features.
set_features	10	1'b0	Операция Set Features.
read_unique_id	11	1'b0	Операция Read Unique ID.
read_status_enhanced	12	1'b0	Операция Read Status Enhanced.
read_interleaved	13	1'b0	Операция Interleaved/Multiplane чтения.
change_read_column_enhanced	14	1'b0	Операция Change Read Column Enhanced.
copy_back_interleaved	15	1'b0	Операция Copy Back Interleaved.
read_cache_start	16	1'b0	Контроллер выполняет операцию Read.
read_cache_sequential	17	1'b0	Операция Read Cache Sequential.
read_cache_random	18	1'b0	Операция Read Cache Random.
read_cache_end	19	1'b0	Операция Read Cache End.
small_data_move	20	1'b0	Операция Small Data Move.
change_row_addr	21	1'b0	Операция Change Row Address.
change_row_addr_end	22	1'b0	1'b1 - завершение Change Row Address.
reset_lun	23	1'b0	1'b1 - операция сброса заданного LUN
pgm_pg_reg_clr	24	1'b0	1'b1 - операция с включённым свойством Page Register Clear Enhancement.
volume_select	25	1'b0	Операция Volume Select
odt_configure	26	1'b0	Операция ODT Configure
–	31:27	5'd0	Зарезервировано.

Примечание. Допускается устанавливать только один разряд в регистре PROGRAM (см. исключения ниже). В случае одновременной установки нескольких разрядов поведение NAND предсказать невозможно.

Примечание. Операции, для которых допускается установка двух разрядов в регистре PROGRAM:

Multi die Page Program. Установлены multi_die и page_program.

Multi die Read. Установлены разряды multi_die и read.

Multi die Block Erase. Установлены multi_die и block_erase.

Change Row Address. В начале выполнения команды необходимо установить разряды change_row_addr и page_program.

Для устройств с включённым свойством "Program Page Register Clear Enhancement" используется увеличенное время tADL. Для использования увеличенного времени tADL

необходимо установить pgm_pg_reg_clr. Время применяется только к последовательностям команд Program (80h).

Примечание. Для команд, запущенных битами read и read_cache_start, выполняется обычная операция чтения. Для команды чтения, запущенной разрядом read, прерывание “Transfer Complete” установится после считывания всех байтов данных из флэш-памяти. Для команды чтения, запущенной разрядом read_cache_start, прерывание “Transfer Complete” установится после получения готовности от флэш-памяти к считыванию данных из неё.

Примечание. Разряд page_program используется для следующих команд:

1. Page Program (80h-10h).
2. Page Program Interleaving (80h-11h).
3. Page Cache Program (80h-15h).

29.2.7 Регистр INTERRUPT_STATUS_EN (0x14)

Таблица 29.9. INTERRUPT_STATUS_EN — регистр разрешение статуса прерываний

Поле	Биты	Значение по сбросу	Описание
buff_wr_rdy_sts_en	0	1’b0	Разрешение признака прерывания “Buffer Write Ready” в регистре INTERRUPT_STATUS (разряд 0). 1'b1 – разрешено. 1'b0 – запрещено.
buff_rd_rdy_sts_en	1	1’b0	Разрешение признака прерывания “Buffer Read Ready” в регистре INTERRUPT_STATUS (разряд 1). 1'b1 – разрешено. 1'b0 – запрещено.
trans_comp_sts_en	2	1’b0	Разрешение признака прерывания “Transfer Complete” в регистре INTERRUPT_STATUS (разряд 2). 1'b1 – разрешено. 1'b0 – запрещено.
mul_bit_err_sts_en	3	1’b0	Разрешение признака прерывания “Multi Bit Error” в регистре INTERRUPT_STATUS (разряд 3). 1'b1 – разрешено. 1'b0 – запрещено. Значение данного поля используется при коррекции ошибок по алгоритму Хемминга (HAM), иначе заменяется нулем.
err_intrpt_sts_en	4	1’b0	Разрешение признака прерывания “Single Bit Error” в режиме HAM или “BCH Detect Error” в режиме BCH в регистре INTERRUPT_STATUS (разряд 4). 1'b1 – разрешено. 1'b0 – запрещено.
ecc_err_intrpt_sts_en	5	1’b0	Разрешение признака прерывания “ECC Error” в регистре INTERRUPT_STATUS (разряд 5) для режима SLC (Хемминг). 1'b1 – разрешено. 1'b0 – запрещено.

Поле	Биты	Значение по сбросу	Описание
dma_int_sts_en	6	1'b0	Разрешение признака прерывания "dma_int" в регистре INTERRUPT_STATUS (разряд 6). 1'b1 – разрешено. 1'b0 – запрещено. Используется только при пересылке в режиме MDMA, иначе заменяется нулем.
error_sys_sts_en	7	1'b0	Разрешение признака прерывания "error_sys" в регистре INTERRUPT_STATUS (разряд 7). 1'b1 – разрешено. 1'b0 – запрещено. Используется только при пересылке в режиме MDMA, иначе заменяется нулем.
–	31:8	24'h0	Зарезервировано.

29.2.8 Регистр INTERRUPT_SIGNAL_EN (0x18)

Таблица 29.10. INTERRUPT_SIGNAL_EN — регистр разрешения сигнала прерываний

Поле	Биты	Значение по сбросу	Описание
buff_wr_rdy_sig_en	0	1'b0	Разрешение установки сигнала прерывания по событию «Buffer Write Ready». 1'b1 – разрешено. 1'b0 – запрещено.
buff_rd_rdy_sig_en	1	1'b0	Разрешение установки сигнала прерывания по событию «Buffer Read Ready». 1'b1 – разрешено. 1'b0 – запрещено.
trans_comp_sig_en	2	1'b0	Разрешение установки сигнала прерывания по событию «Transfer Complete». 1'b1 – разрешено. 1'b0 – запрещено.
mul_bit_err_sig_en	3	1'b0	Разрешение установки сигнала прерывания по событию «Multi Bit Error». 1'b1 – разрешено. 1'b0 – запрещено. Значение данного поля используется при коррекции ошибок по алгоритму Хемминга (HAM), иначе заменяется нулем.
err_intrpt_sig_en	4	1'b0	Разрешение установки сигнала прерывания по событию «Single Bit Error» в режиме HAM или «BCH Detect Error» в режиме BCH. 1'b1 – разрешено. 1'b0 – запрещено.
ecc_err_intrpt_sig_en	5	1'b0	Разрешение установки сигнала прерывания по событию «ECC Error» для режима SLC (Хемминг). 1'b1 – разрешено. 1'b0 – запрещено.
dma_int_sig_en	6	1'b0	Разрешение установки сигнала прерывания по событию «dma_int». 1'b1 – разрешено. 1'b0 – запрещено. Используется только при пересылке в режиме MDMA, иначе заменяется нулём.

Поле	Биты	Значение по сбросу	Описание
error_sys_sig_en	7	1’b0	Разрешение установки сигнала прерывания по событию «error_sys». 1'b1 – разрешено. 1'b0 – запрещено. Используется только при пересылке в режиме MDMA, иначе заменяется нулём.
–	31:8	24’h0	Зарезервировано.

29.2.9 Регистр INTERRUPT_STATUS (0x1C)

Таблица 29.11. INTERRUPT_STATUS — регистр статуса прерываний

Поле	Биты	Значение по сбросу	Описание
buff_wr_rdy_reg	0	1’b0	«Buffer Write Ready». Готовность FIFO к приёму данных от процессора. Устанавливается при условии, что в буфере достаточно места для получения блока данных от процессора.
buff_rd_rdy_reg	1	1’b0	«Buffer Read Ready». Готовность FIFO к передаче данных процессору. Устанавливается при условии, что считано достаточно данных из флэш-памяти.
trans_comp_reg	2	1’b0	«Transfer Complete». Пересылка завершена. Устанавливается после завершения выполнения заданной команды.
mul_bit_err_reg	3	1’b0	«Multi Bit Error». Устанавливается при условии, что подтверждается ошибка в нескольких битах. Используется только в режиме HAM, иначе считается нулём.
err_intrpt_reg	4	1’b0	«Single Bit Error»/«BCH Detect Error». Устанавливается при условии, что обнаруживается ошибка BCH или одиночная ошибка в HAM режиме.
ecc_err_intrpt_reg	5	1’b0	«ECC Error». Устанавливается при обнаружении однобитной ошибки в области ECC для HAM режима.
dma_int_reg	6	1’b0	«dma_int». Установка бита происходит при достижении границы буфера в режиме MDMA. Используется только при пересылке в режиме MDMA, иначе заменяется нулём.
error_sys_reg	7	1’b0	«error_sys». Устанавливается при выполнении пересылки данных на системной шине в режиме MDMA, если получен отклик с признаком ошибки. Используется только при пересылке в режиме MDMA, иначе заменяется нулём.
–	31:8	24’h0	Зарезервировано.

Примечание. Для сброса статуса прерывания нужно записать единицу в необходимый разряд регистра INTERRUPT_STATUS.

29.2.10 Регистр READY_BUSY (0x20)

Таблица 29.12. READY_BUSY - Регистр готовности флэш

Поле	Биты	Значение по сбросу	Описание
rb_n0	0	1’b1	Ready/Busy флэш 0.
rb_n1	1	1’b1	Ready/Busy флэш 1.

–	31:2	30'h0	Зарезервировано.

29.2.11 Регистр FLASH_STATUS (0x28)

Хранит результат выполнения команды Read Status.

Таблица 29.13. FLASH_STATUS — регистр статуса флэш-памяти

Поле	Биты	Значение по сбросу	Описание
flash_status	15:0	16'h0	Значение статуса флэш-памяти.
–	31:16	16'h0	Зарезервировано

Примечание. При выполнении команды Read Status результат команды записывается в регистр FLASH_STATUS, устанавливается прерывание «Transfer Complete». Только после этого процессору следует прочитать регистр FLASH_STATUS.

Примечание. При последовательном способе работы с флэш-памятью:

Разряды FLASH_STATUS [7:0] содержат значение статуса подключённого флэш-устройства.

Разряды FLASH_STATUS [15:8] установлены в ноль.

В параллельном режиме работы с флэш-памятью:

1. Разряды FLASH_STATUS [7:0] содержат значение статуса значение статуса первого флэш-устройства.
2. Разряды FLASH_STATUS [15:8] содержат значение статуса значение статуса второго флэш-устройства.

29.2.12 Регистр TIMING (0x2C)

Таблица 29.14. TIMING - Регистр временных параметров

Поле	Биты	Значение по сбросу	Описание
tccs_time	1:0	2'd0	tccs (время установки после смены столбца, см. ONFI). 2'b00 - 500 ns 2'b01 - 100 ns 2'b10 - 200 ns 2'b11 - 300 ns Поле tccs time настраивается на основе значений байтов [139:140] страницы параметров.
slow_fast_tcad	2	1'b0	tcad (Command, Address, Data delay, см. ONFI). 0 – медленное устройство (tCADs). 1 – быстрое устройство (tCADf). Поле slow_fast_tcad настраивается на основе значения байта [143] страницы параметров.
dqs_buff_sel_in	6:3	4'h0	Задержка данных при чтении. Для чтения необходимо настроить на основании значения tDQSQ.

Поле	Биты	Значение по сбросу	Описание
			NVDDR Mode 0 – 4'h6 NVDDR Mode 1 – 4'h4 NVDDR Mode 2 – 4'h3 NVDDR Mode 3 – 4'h3 NVDDR Mode 4 – 4'h2 NVDDR Mode 5 – 4'h2 NVDDR2 Mode 0 – 4'h5 NVDDR2 Mode 1 – 4'h4 NVDDR2 Mode 2 – 4'h3 NVDDR2 Mode 3 – 4'h2 NVDDR2 Mode 4 – 4'h2 NVDDR2 Mode 5 – 4'h1 NVDDR2 Mode 6 – 4'h1 NVDDR2 Mode 7 – 4'h1 NVDDR2 Mode 8 – 4'h1
tadl_time	14:7	7'd0	tadl (см. ONFI). Время между циклом адреса и загрузкой данных. Данное поля должно быть запрограммировано при операции программирования флэш-памяти, если установлен бит pgm_pg_reg_clr. Значения поля tadl_time определяется значениями байтов [154:155] страницы параметров.
dqs_buff_sel_out	18:15	4'h0	Задержка данных при записи. Для записи необходимо настроить на основании значения tDS. NVDDR Mode 0 – 4'h6 NVDDR Mode 1 – 4'h4 NVDDR Mode 2 – 4'h3 NVDDR Mode 3 – 4'h3 NVDDR Mode 4 – 4'h2 NVDDR Mode 5 – 4'h2 NVDDR2 Mode 0 – 4'h5 NVDDR2 Mode 1 – 4'h4 NVDDR2 Mode 2 – 4'h3 NVDDR2 Mode 3 – 4'h2 NVDDR2 Mode 4 – 4'h2 NVDDR2 Mode 5 – 4'h1 NVDDR2 Mode 6 – 4'h1 NVDDR2 Mode 7 – 4'h1 NVDDR2 Mode 8 – 4'h1
–	31:19	25'h0	Зарезервировано.

29.2.13 Регистр BUFFER_DATA (0x30)

Таблица 29.15. BUFFER_DATA - регистр данных буфера

Поле	Биты	Значение по сбросу	Описание
data_port	31:0	32'h0	Порт доступа к данным внутреннего FIFO (буфера). Через данный регистр производится чтение/запись FIFO контроллера.

29.2.14 Регистр ECC (0x34)

Регистр используется при включённой возможности обнаружения и коррекции ошибок. Регистр настраивает адрес и размер области флэш-памяти, где находятся данные ECC, а также способ коррекции.

Таблица 29.16. ECC - регистр ECC

Поле	Биты	Значение по сбросу	Описание
ecc_addr	15:0	16'h00	Адрес столбца (Column Address) страницы флэш-памяти, определяющий начальное местоположение кодов коррекции. Используется для формирования адреса ECC с командами Change Read Column, Change Write Column. Рекомендуется выбирать значение ecc_addr = *размер страницы в байтах* + *размер spare-области* - ecc_size.
ecc_size	26:16	11'h00	Задаёт количество транзакций записи или чтения для области памяти с данными ECC.
ham_bch	27	1'b0	Выбор способа коррекции. 1'b0 - HAM 1'b1 - BCH
–	31:28	4'h0	Зарезервировано.

Таблица 29.17. Значение поля ecc_size в зависимости от размера страницы и режима ECC (последовательный режим)

Размер Страницы\Режим ECC	HAM	BCH-4	BCH-8	BCH-12	BCH-16	BCH-24
512	9'h 003	9'h 007	9'h 00D	-	-	-
2k	9'h 00C	9'h 01A	9'h 034	9'h 04E	9'h 068	9'h 054
4k	9'h 018	9'h 034	9'h 068	9'h 09C	9'h 0D0	9'h 0A8
8k	9'h 030	9'h 068	9'h 0D0	9'h 138	9'h 1A0	9'h150

Таблица 29.18. Значение поля ecc_size в зависимости от размера страницы и режима ECC (параллельный режим)

Размер Страницы\Режим ECC	HAM	BCH-4	BCH-8	BCH-12	BCH-16	BCH-24
512	9'h 006	9'h 00E	9'h 00A	-	-	-
2k	9'h 018	9'h 034	9'h 068	-	9'h 0D0	9'h 0A8
4k	9'h 030	9'h 068	9'h 0D0	-	9'h 1A0	9'h 150
8k	9'h 060	9'h 0D0	9'h 1A0	-	-	-

29.2.15 Регистр ECC_ERROR_CNT (0x38)

Таблица 29.19. ECC_ERROR_CNT — счётчик ошибок ECC

Поле	Биты	Значение по сбросу	Описание
packet_bound_err_count	7:0	8'b0	Счётчик ошибок при чтении пакета данных. ПО должно считывать значение данного поля каждый раз после чтения данных размером packet_size.

page_bound_err_count	16:8	9'b0	Общий счётчик ошибок при чтении страницы. ПО должно считывать значение данного поля после прочтения каждой страницы.
–	31:17	15'h0	Зарезервировано

Содержимое данного регистра может быть считано при обработке прерывания "Transfer Complete" для следующих операций: read, multi_die_rd, change_read_column_enhanced, read_cache_sequential, read_cache_random, read_cache_end.

29.2.16 Регистр ECC_SPARE_CMD (0x3C)

Таблица 29.20. ECC_SPARE_CMD - регистр команд ECC

Поле	Биты	Значение по сбросу	Описание
ecc_spare_cmd	15:0	16'h 00	Используется для программирования кода операций ECC, которые выполняются совместно с командами Change Write Column, Change Read Column.
reserved	27:16	12'h 00	Зарезервировано
ecc_spare_num_addr_cycles	30:28	3'b000	Число циклов адреса в команде ECC для флэш-памяти. 3'b000 – Не используется 3'b001 – Один цикл ... 3'b111 — Семь циклов
–	31	1'b0	Зарезервировано

29.2.17 Регистры ERROR_CNT_nBIT

Регистр ERROR_CNT_1BIT — счётчик 1-битных ошибок (0x40).

Регистр ERROR_CNT_2BIT — счётчик 2-битных ошибок (0x44).

Регистр ERROR_CNT_3BIT — счётчик 3-битных ошибок (0x48).

Регистр ERROR_CNT_4BIT — счётчик 4-битных ошибок (0x4C).

Регистр ERROR_CNT_5BIT — счётчик 5-битных ошибок (0x5C).

Регистр ERROR_CNT_6BIT — счётчик 6-битных ошибок (0x60).

Регистр ERROR_CNT_7BIT — счётчик 7-битных ошибок (0x64).

Регистр ERROR_CNT_8BIT — счётчик 8-битных ошибок (0x68).

Таблица 29.21. ERROR_CNT_nBIT — счётчик n-битных ошибок (n от 1 до 8)

Поле	Биты	Значение по сбросу	Описание
error_count_n_bit	31:0	32'h 00	Количество обнаруженных n-битных ошибок. При необходимости сбрасывается ПО.

29.2.18 Регистр DATA_INTF (0x6C)

Таблица 29.22. DATA_INTF – настройка режима работы интерфейса флэш

Поле	Биты	Значение по сбросу	Описание
sdr	2:0	3'b000	3'b000 - SDR mode 0 3'b001 - SDR mode 1 3'b010 - SDR mode 2 3'b011 - SDR mode 3 3'b100 - SDR mode 4 3'b101 - SDR mode 5 3'b110 - зарезервировано 3'b111 - зарезервировано
nvddr	5:3	3'b000	3'b000 - NVDDR mode 0 3'b001 - NVDDR mode 1 3'b010 - NVDDR mode 2 3'b011 - NVDDR mode 3 3'b100 - NVDDR mode 4 3'b101 - NVDDR mode 5 3'b110 - зарезервировано 3'b111 - зарезервировано
nvddr2	9:6	4'b0000	4'b0000 - NVDDR2 mode 0 4'b0001 - NVDDR2 mode 1 4'b0010 - NVDDR2 mode 2 4'b0011 - NVDDR2 mode 3 4'b0100 - NVDDR2 mode 4 4'b0101 - NVDDR2 mode 5 4'b0110 - NVDDR2 mode 6 4'b0111 - NVDDR2 mode 7 4'b1000 - NVDDR2 mode 8
data_intf	11:10	2'b00	2'b00 - SDR 2'b01 - NVDDR 2'b10 - NVDDR2
–	31:12	20'b0	Зарезервировано.

В начальный момент флэш работает в режиме SDR mode 0. ПО должно записать 0 в регистр DATA_INTF. После смены временного режима (команда set feature) необходимо перенастроить содержимое данного регистра в соответствии с новым режимом.

29.2.19 Регистр DMA_SYS_ADDR0 (0x50)

Таблица 29.23. DMA_SYS_ADDR0 – системный адрес 0 DMA

Поле	Биты	Значение по сбросу	Описание
dma_sys_addr	31:0	32'h 00	Используется только в режиме MDMA. Содержит младшую часть адреса системной памяти для DMA-передачи при использовании 64-битной адресации. Драйвер должен проинициализировать регистр до запуска DMA. При достижении границы, заданной в регистре DMA_BUFFER_BOUNDARY, контроллер генерирует прерывание. Процессор должен обработать этот запрос и настроить адрес следующей транзакции.

29.2.20 Регистр DMA_SYS_ADDR0 (0x24)

Таблица 29.24. DMA_SYS_ADDR1 — системный адрес 1 DMA

Поле	Биты	Значение по сбросу	Описание
dma_sys_addr	31:0	32’h 00	Используется только в режиме MDMA при использовании 64-битной адресации. Содержит старшую часть адреса системной памяти.

29.2.21 Регистр DMA_BUFFER_BOUNDARY (0x54)

Таблица 29.25. DMA_BUFFER_BOUNDARY - граница DMA пересылки

Поле	Биты	Значение по сбросу	Описание
dma_buffer_size	2:0	3’h0	Используется только в режиме MDMA. Для выполнения больших пересылок, регистр DMA_SYS_ADDR должен обновляться процессором при достижении границы, заданной в регистре DMA_BUFFER_BOUNDARY. Данное поле определяет размер непрерывного буфера данных в системной памяти: 3'b000 - 4KB 3'b001 - 8KB 3'b010 - 16KB 3'b011 - 32KB 3'b100 - 64KB 3'b101 - 128KB 3'b110 - 256KB 3'b111 - 512KB Достижение границы для каждого из размеров буфера определяется по изменению соответствующего разряда адреса (12, 13, 14, 15, 16, 17, 18 или 19-ого) .

29.2.22 Регистр CPU_RELEASE (0x58)

Таблица 29.26. CPU_STATE — Регистр CPU_RELEASE (0x58)

Поле	Биты	Значение по сбросу	Описание
cpu_release	1	1'b0	Устанавливается в единицу, если загрузка кода CPU была выполнена из NAND флэш-памяти.
–	31:2	31'b0	Зарезервировано.

29.3 Описание команд

В данном разделе описаны примеры выполнения различных операций с флэш-памятью.

29.3.1 Схема выполнения команды «Без доступа к FIFO»

Для команды, не требующей от NAND пересылок данных через FIFO, обычно выполняется следующая последовательность действий:

Пункты 1-7 – инициирование команды

1. CPU разрешает формирование признака прерывания “Transfer Complete” (см. регистр INTERRUPT_STATUS_EN).
2. CPU разрешает формирование сигнала прерывания (см. регистр INTERRUPT_SIGNAL_EN).
3. CPU настраивает регистр COMMAND.
4. CPU настраивает регистр MEMADDR1.
5. CPU настраивает регистр MEMADDR2.
6. CPU инициирует выполнение команды установкой необходимого бита в регистре PROGRAM.
7. NAND выдаёт команду флэш-устройству.

   Пункты 8-14 — завершение отработки команды

8. CPU ожидает прерывание “Transfer Complete”.
9. По готовности флэш-памяти и завершении выполнения команды NAND устанавливает признак прерывания “Transfer Complete” в регистре INTERRUPT_STATUS.
10. Формируется физический сигнал прерывания.
11. CPU считывает регистр INTERRUPT_STATUS.
12. CPU сбрасывает бит “Transfer Complete” регистра INTERRUPT_SIGNAL_EN.
13. CPU сбрасывает бит “Transfer Complete” регистра INTERRUPT_STATUS_EN.
14. CPU сбрасывает бит “Transfer Complete” в регистре INTERRUPT_STATUS.

Можно использовать альтернативный метод работы, не используя сигнал прерывания. В этом случае разрешать формирование физического сигнала прерывания в регистре INTERRUPT_SIGNAL_EN не следует. Вместо этого CPU должен опрашивать регистр INTERRUPT_STATUS и проверять состояние разряда “Transfer Complete”.

29.3.2 Схема выполнения команды «С чтением из FIFO»

Для команды, требующей от CPU чтения из FIFO обычно выполняется следующая последовательность действий (пункты 1-8 – инициирование команды):

1. CPU разрешает формирование признака прерывания «Buffer Read Ready» (см. регистр INTERRUPT_STATUS_EN).

2. CPU разрешает формирование сигнала прерывания «Buffer Read Ready» (см. регистр INTERRUPT_SIGNAL_EN).
3. CPU настраивает регистр COMMAND.
4. CPU настраивает регистр MEMADDR1.
5. CPU настраивает регистр MEMADDR2.
6. CPU настраивает регистр PACKET.
7. При включённых возможностях ECC (автоматическом обнаружении и коррекции ошибок) CPU должен настроить содержимое регистров ECC и ECC_SPARE_CMD.
8. CPU инициирует выполнение команды установкой необходимого бита в регистре PROGRAM.
9. NAND выдаёт команду флэш-устройству и начинает считывать из него данные.

   Пункты 10-21 — считывание данных из FIFO.

10. CPU ожидает прерывание “Buffer Read Ready”.
11. После того как в FIFO появляется хотя бы один доступный пакет данных в регистре INTERRUPT_STATUS устанавливается признак прерывания “Buffer Read Ready”.
12. Формируется физический сигнал прерывания.
13. CPU считывает регистр INTERRUPT_STATUS.
14. CPU запрещает формирование сигнала прерывания “Buffer Read Ready” (см. регистр INTERRUPT_SIGNAL_EN).
15. Если число прерываний равно числу переданных пакетов, выполняется шаг 16, иначе шаг 17.
16. CPU разрешает формирование признака прерывания “Transfer Complete” (см. регистр INTERRUPT_STATUS_EN). Далее шаг 18.
17. CPU сбрасывает бит “Buffer Read Ready” регистра INTERRUPT_STATUS_EN.
18. CPU сбрасывает бит “Buffer Read Ready” регистра INTERRUPT_STATUS.
19. CPU считывает пакет данных через регистр BUFFER_DATA.
20. Если число прерываний равно числу переданных пакетов, выполняется шаг 23, иначе шаг 21.

21. CPU разрешает формирование признака прерывания “Buffer Read Ready” (см. регистр INTERRUPT_STATUS_EN).

22. CPU разрешает формирование сигнала прерывания “Buffer Read Ready” (см. регистр INTERRUPT_SIGNAL_EN). Далее шаг 10.

    Пункты 23-30 — завершение отработки команды.

23. CPU разрешает формирование сигнала прерывания “Transfer Complete” (см. регистр INTERRUPT_SIGNAL_EN).

24. CPU ожидает прерывание “Transfer Complete”.

25. После завершения пересылки всех байтов данных для CPU NAND устанавливает признак прерывания “Transfer Complete” в регистре INTERRUPT_STATUS.

26. Формируется физический сигнал прерывания.

27. CPU считывает регистр INTERRUPT_STATUS.

28. CPU сбрасывает бит “Transfer Complete” регистра INTERRUPT_SIGNAL_EN.

29. CPU сбрасывает бит “Transfer Complete” регистра INTERRUPT_STATUS_EN.

30. CPU сбрасывает бит “Transfer Complete” в регистре INTERRUPT_STATUS.

Можно использовать альтернативный метод работы, не используя сигнал прерывания. В этом случае разрешать формирование физического сигнала прерывания в регистре INTERRUPT_SIGNAL_EN не следует. Вместо этого CPU должен опрашивать регистр INTERRUPT_STATUS и проверять состояние разрядов “Transfer Complete”, “Buffer Read Ready”.

29.3.3 Схема выполнения команды «С записью в FIFO»

Для команды, требующей от CPU записи в FIFO, обычно выполняется следующая последовательность действий:

Пункты 1-8 – инициирование команды

1. CPU разрешает формирование признака прерывания “Buffer Write Ready” (см. регистр INTERRUPT_STATUS_EN).

2. CPU разрешает формирование сигнала прерывания “Buffer Write Ready” (см. регистр INTERRUPT_SIGNAL_EN).

3. CPU настраивает регистр COMMAND.

4. CPU настраивает регистр PACKET.
5. CPU настраивает регистр MEMADDR1.
6. CPU настраивает регистр MEMADDR2.
7. При включённых возможностях ECC (автоматическом обнаружении и коррекции ошибок) CPU должен настроить содержимое регистров ECC и ECC_SPARE_CMD.
8. CPU инициирует выполнение команды установкой необходимого бита в регистре PROGRAM.
9. NAND выдаёт команду флэш-устройству и начинает выполнять её.

   Пункты 10-22 – запись данных в FIFO.

10. CPU ожидает прерывание “Buffer Write Ready”.
11. Если в FIFO достаточно места для приёма данных от CPU, в регистре INTERRUPT_STATUS устанавливается признак прерывания “Buffer Write Ready”.
12. Формируется физический сигнал прерывания.
13. CPU считывает регистр INTERRUPT_STATUS.
14. CPU запрещает формирование сигнала прерывания “Buffer Write Ready” (см. регистр INTERRUPT_SIGNAL_EN).
15. Если число прерываний равно числу переданных пакетов, выполняется шаг 16, иначе шаг 17.
16. CPU разрешает формирование признака прерывания “Transfer Complete” (см. регистр INTERRUPT_STATUS_EN). Далее шаг 18.
17. CPU сбрасывает бит “Buffer Write Ready” регистра INTERRUPT_STATUS_EN.
18. CPU сбрасывает бит “Buffer Write Ready” регистра INTERRUPT_STATUS.
19. CPU записывает пакет данных в FIFO через регистр BUFFER_DATA.
20. Если число прерываний равно числу переданных пакетов, выполняется шаг 23, иначе шаг 21.
21. CPU разрешает формирование признака прерывания “Buffer Write Ready” (см. регистр INTERRUPT_STATUS_EN).

22. CPU разрешает формирование сигнала прерывания “Buffer Write Ready” (см. регистр INTERRUPT_SIGNAL_EN). Далее шаг 10.

    Пункты 23-31 – завершение отработки команды.

23. После того как все байты данных будут записаны во флэш-память NAND завершает выполнение команды.

24. CPU разрешает формирование сигнала прерывания “Transfer Complete” (см. регистр INTERRUPT_SIGNAL_EN).

25. CPU ожидает прерывание “Transfer Complete”.

26. После завершения пересылки всех байтов данных NAND устанавливает признак прерывания “Transfer Complete” в регистре INTERRUPT_STATUS.

27. Формируется физический сигнал прерывания.

28. CPU считывает регистр INTERRUPT_STATUS.

29. CPU сбрасывает бит “Transfer Complete” регистра INTERRUPT_SIGNAL_EN.

30. CPU сбрасывает бит “Transfer Complete” регистра INTERRUPT_STATUS_EN.

31. CPU сбрасывает бит “Transfer Complete” в регистре INTERRUPT_STATUS.

Можно использовать альтернативный метод работы, не используя сигнал прерывания. В этом случае разрешать формирование физического сигнала прерывания в регистре INTERRUPT_SIGNAL_EN не следует. Вместо этого CPU должен опрашивать регистр INTERRUPT_STATUS и проверять состояние разрядов “Transfer Complete”, “Buffer Write Ready”.

29.3.4 Команда Reset

Команда Reset (FFh) должна быть первой командой после включения питания флэш-памяти.

Команда Reset выполняется по схеме «Без доступа к FIFO», со следующими особенностями:

- настройку регистра MEMADDR1 можно не производить;
- в регистр COMMAND записывается значение 0xFF;
- в регистре PROGRAM необходимо установить бит reset.

29.3.5 Команда Read Status

Команда Read Status используется для получения статуса последней выполненной операции.

Данная команда выполняется по схеме «Без доступа к FIFO», со следующими особенностями:

1. Настройку регистра MEMADDR1 можно не производить.
2. Перед запуском команды через регистр PROGRAM необходимо дополнительно настроить регистр PACKET.
3. В регистре PROGRAM необходимо установить бит read_status.

После выполнения команды процессор может произвести чтение регистра FLASH_STATUS (данные о состоянии последней операции были загружены из флэш-памяти в этот регистр в результате выполнения команды).

29.3.6 Команда Read ID

С помощью команды Read ID определяется соответствие целевого устройства стандарту ONFI.

Команда Read ID с адресом 20h – соответствие стандарту ONFI.

Команда Read ID с адресом 00h – JEDEC ID.

Данная команда выполняется по схеме «Без доступа к FIFO», со следующими особенностями:

1. Перед запуском команды через регистр PROGRAM необходимо дополнительно настроить регистр PACKET.
2. В регистре PROGRAM необходимо установить бит read_id.

После выполнения команды процессор может произвести чтение регистров ID1 и ID2 (данные из флэш-памяти были загружены в эти регистры в результате выполнения команды). При выполнении команды Read ID с адресом 20h (ONFI ID) считанные из флэш-памяти четыре байта сохраняются в {ID2[7:0], ID1[31:8]}. При выполнении команды Read ID с адресом 00h (JEDEC ID) считанные из флэш-памяти пять байтов сохраняются в {ID2[7:0], ID1[31:0]}.

29.3.7 Команда Read Parameter Page

Команда Read Parameter Page получает информацию об организации, характеристиках, временных параметрах и другие особенностях целевого устройства.

Данная команда выполняется по схеме «С чтением из FIFO», со следующей особенностью:

в регистре PROGRAM необходимо установить бит read_parameter_page.

29.3.8 Команда Page Program без ECC

Данная команда осуществляет передачу страницы данных или части страницы данных, определяемой адресом столбца, в регистр страницы. Содержимое регистра страницы затем пересылается в массив флэш-памяти по заданному адресу строки.

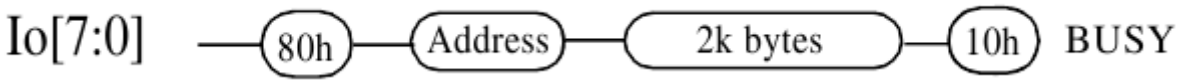


Рисунок 29.2. Команда Page Program (без ECC)

Данная команда выполняется по схеме «Команды с записью в FIFO», со следующей особенностью: в регистре PROGRAM установлен бит page_program.

При последовательном способе работы с флэш-памятью общее число пересылок (packet_size x packet_count) не должно превышать размер страницы флэш-памяти. Для параллельного режима (одновременно с двумя флэш-памятями) максимальный размер пересылок может быть в два раза больше размера страницы, так как поле packet_count для параллельного режима должно быть в два раза больше, чем для последовательного.

Пример.

При последовательном режиме работы и размере страницы 8k можно выбрать следующие значения полей регистра PACKET:

packet_size = 512, packet_count = 16.

В параллельном режиме работы и размере страницы 8k можно выбрать следующие значения полей регистра PACKET:

packet_size = 512, packet_count = 32.

29.3.9 Команда Page Program с ECC

Данная команда осуществляет передачу страницы или части страницы данных, определяемой адресом столбца, в регистр страницы. Содержимое регистра страницы затем пересылается в массив флэш-памяти по заданному адресу строки.

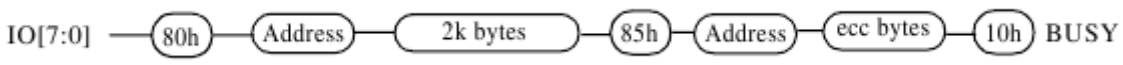


Рисунок 29.3. Команда Page Program (с ECC)

Данная команда выполняется по схеме «С записью в FIFO», со следующими особенностями:

1. В регистре PROGRAM установлен бит page_program.
2. В регистре COMMAND установлен бит ECC_ON_OFF
3. В регистре ECC:
    - Для HAM поле ham_bch = 0, ecc_addr = 0x834, ecc_size = 0x0c.
    - Для BCH поле ham_bch = 1, ecc_addr = 0x826, ecc_size = 0x1a.
4. В регистре ECC_SPARE_CMD:
    - Поле ecc_spare_cmd = 0x0085 (Change Write Column).
    - Поле ecc_spare_num_addr_cycles = 0x02.

Байты данных, записанные в FIFO для пересылки во флэш-память, подаются на вход модуля ECC для вычисления контрольных разрядов ECC. Контрольные разряды вычисляются для каждой последовательности из 512 байт.

После того как все пакеты данных переданы во флэш-память, NAND выполняет команду 0x85 (Change Write Column) с двумя адресными циклами. После чего NAND записывает вычисленные контрольные разряды в резервную область флэш-памяти.

Примечание. Значения поля ecc_addr 0x834 или 0x826 используется для страницы размером 2k. Данные значения выбраны таким образом, чтобы записать контрольные разряды ECC в последние свободные резервные (spare) байты данной страницы.

Адрес контрольных разрядов для страницы размером 2k в режиме HAM = общее количество байтов в странице – длина кода ECC = 0x840 - 0x0C = 0x834.

Адрес контрольных разрядов для страницы размером 2k в режиме BCH = общее количество байтов в странице – длина кода ECC = 0x840 - 0x0A = 0x826.

29.3.10 Команда Page Program для spare области

Программирование резервной области выполняется аналогично обычной команде программирования. Адрес области для программирования должен указывать на резервную область.

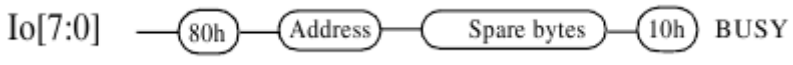


Рисунок 29.4. Команда Page Program (spare)

Таблица 29.27. Рекомендуемые параметры программирования резервной области

Параметр\Размер страницы	2k	4k	8k
Количество байт в spare пересылке	<= 64	<= 128	<= 256
MEMADDR1	> 0x7FF	> 0xFFF	> 0x1FFF

29.3.11 Команда Read без ECC

Команда считывает страницу данных, заданную адресом строки для заданного LUN. Страница данных становится доступной для чтения из регистра страницы, начиная с заданного адреса столбца.

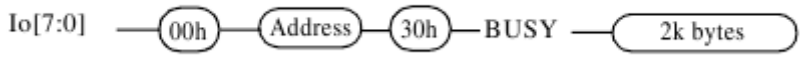


Рисунок 29.5. Команда Чтение (без ECC)

Данная команда выполняется по схеме «С чтением из FIFO», со следующей особенностью: в регистре PROGRAM необходимо установить бит read.

29.3.12 Команда Read с ECC

Команда считывает страницу данных, заданную адресом строки для заданного LUN. Страница данных становится доступной для чтения из регистра страницы, начиная с заданного адреса столбца.

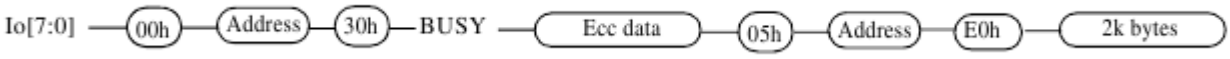


Рисунок 29.6. Команда Чтение (с ECC)

Данная команда выполняется по схеме «С чтением из FIFO», со следующими особенностями:

1. В пунктах 1 и 2 дополнительно разрешить прерывания «Single Bir Error», «Multi Bir Error».
2. В регистре COMMAND (пункт 3) необходимо разрешить возможности ECC (разряд ecc_on_off).
3. Необходимо настроить ECC регистры (пункт 7).
4. Регистр ECC (0x34).
5. Для HAM поле ham_bch = 0, ecc_addr = 0x834, ecc_size = 0x0c.
6. Для BCH поле ham_bch = 1, ecc_addr = 0x826, ecc_size = 0x1a.
7. Регистр ECC_SPARE_CMD (0x3c).
8. Поле ecc_spare_cmd = 0xe005 (Change Read Column).
9. Поле num_of_addr_cycles = 0x02.
10. В регистре PROGRAM (пункт 8) установить разряд read.

Сначала NAND выполнит команду Read с адресом, заданным в регистре ECC, и произведёт считывание контрольных байтов. После этого NAND выполнит команду Change Read Column с двумя циклами адреса и приступит к считыванию данных из основной области.

Данные, считанные из основной области флэш-памяти в FIFO, подаются на вход блока ECC для вычисления контрольных разрядов ECC. Контрольные разряды ECC вычисляются для каждого блока из 512 байт. Блок ECC осуществляет проверку вычисленного кода ECC со считанным из флэш-памяти.

Настройка полей ecc_size, ecc_addr регистра ECC выполняется аналогично команде Page Program (с ECC).

29.3.13 Команда Read для spare области

Чтение из резервной области выполняется аналогично обычной команде чтения. Адрес считываемой области должен указывать на резервную область.

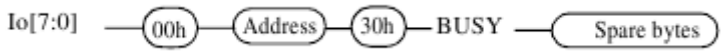


Рисунок 29.7. Чтение из резервной области

Ограничения на MEMADDR1 и размер пересылок аналогично команде Page Program для spare области.

29.3.14 Команда Block Erase

Команда Block Erase стирает блок данных, определяемый адресом блока в заданном логическом устройстве (LUN).



Рисунок 29.8. Команда Block Erase

Данная команда выполняется по схеме «Без доступа к FIFO», со следующими особенностями:

- COMMAND[15:0] = 0xD060;
- в регистре PROGRAM необходимо установить бит block_erase.

29.3.15 Команда Multi-plane Page Program

Команда Multi-plane Page Program (80h-11h) позволяет производить одновременную пересылку нескольких регистров страниц в различные блоки массива флэш-памяти, что позволяет увеличить производительность системы. Для того чтобы выполнить операцию программирования одновременно над несколькими блоками, необходимо перед командой Page Program (80h-10h) добавить одну или несколько команд Multi-plane Page Program.

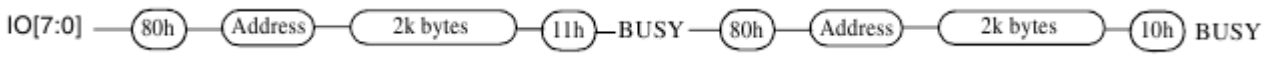


Рисунок 29.9. Multi-plane Page Program

Пример использования Multi-plane Page Program:

1. Команда Multi-plane Page Program выполняется по схеме «С записью в FIFO» со следующими особенностями:

COMMAND[15:0] = 0x1180.

В регистре PROGRAM необходимо установить бит page_program.

2. Команда Page Program выполняется по схеме «С записью в FIFO», со следующими особенностями:

COMMAND[15:0] = 0x1080.

В регистре PROGRAM необходимо установить бит page_program.

29.3.16 Команды Multi-plane Copyback

Обычная операция копирования памяти состоит из двух шагов:

1. Выполняется команда Copyback Read.

Команда загружает заданную страницу в регистр страницы.

2. Выполняется команда Copyback Program

Команда программирует заданную страницу содержимым регистра страницы.

Для увеличения производительности можно использовать команды Multi-plane Read и Multi-plane Copyback Program.

Чтобы выполнить одновременное копирование нескольких страниц в различные регистры страниц, необходимо перед командой Copyback Read (00h-35h) добавить одну или несколько команд Multi-plane Read (00h-32h).

Чтобы выполнить одновременное программирование нескольких страниц из различных регистров страниц необходимо перед командой Copyback Program (85h-10h) добавить одну или несколько команд Multi-plane Copyback Program (85h-11h).

Следущая последовательность команд позволяет произвести одновременное копирование двух блоков памяти:



Рисунок 29.10. Одновременное копирование двух блоков в одном LUN

Пример одновременного копирование двух блоков в одном LUN:

1. Команда Read Multi-plane выполняется по схеме «Без доступа к FIFO», со следующими особенностями:

В регистре PROGRAM необходимо установить бит copy_back_interleaved.

COMMAND[15:0] = 0x3200.

2. Команда начинает загрузку заданной страницы в регистр страницы и разрешает выполнение следующей команды, не дожидаясь завершения текущей.

Выполнятся команда Copyback Read по схеме «Без доступа к FIFO», со следующими особенностями:

В регистре PROGRAM необходимо установить бит copy_back_interleaved.

COMMAND[15:0] = 0x3500.

Команда загружает следующую необходимую страницу в другой регистр страницы (определяется адресом блока). Таким образом выполняется параллельное копирование двух разных областей памяти в соответствующие регистры страниц.

3. Выполнятся команда Copyback Program Multi-plane по схеме «Без доступа к FIFO», со следующими особенностями:

В регистре PROGRAM необходимо установить бит copy_back_interleaved.

COMMAND[15:0] = 0x1185.

Команда начинает сохранение регистра страницы в необходимую область памяти и разрешает выполнение следующей команды, не дожидаясь завершения текущей.

4. Выполнятся команда Copyback Program по схеме «Команды без доступа к FIFO», со следующими особенностями:

В регистре PROGRAM установлен бит copy_back_interleaved.

COMMAND[15:0] = 0x1085.

Команда выполняет сохранение другого регистра страницы в необходимую область памяти. Таким образом, выполняется одновременное программирование двух различных областей памяти.

При выполнение multi-plane операций необходимо учитывать ограничения на адреса областей памяти (см. стандарт ONFI).

29.3.17 Команда Multi-plane Read

Команда Read Multi-plane (00h-32h) позволяет производить одновременную загрузку нескольких страниц в различные регистры страниц, что позволяет увеличить производительность системы. Для того чтобы выполнить команду чтения над несколькими блоками одновременно, необходимо перед командой Read (00h-30h) добавить одну или несколько команд Read Multi-plane. Выбор блока для считывания данных из флэш-памяти после её готовности осуществляется командой Change Read Column Enchanced (06h-E0h).

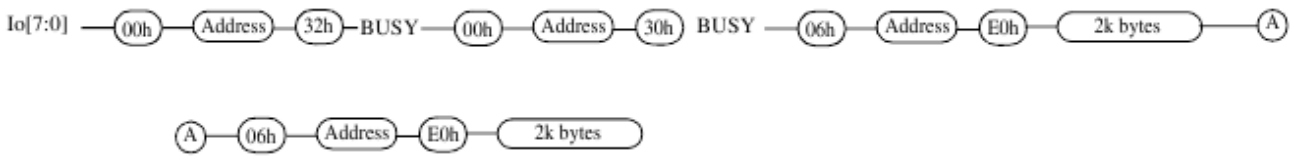


Рисунок 29.11. Ускоренное чтение двух блоков памяти из одного LUN

Ускоренное чтение двух блоков памяти из одного LUN:

1. Выполнятся команда Read Multi-plane по схеме «Без доступа к FIFO» со следующими особенностями:

В регистре PROGRAM установлен бит read_interleaved.

COMMAND[15:0] = 0x3200.

2. Выполнятся команда Read по схеме «Без доступа к FIFO» со следующими особенностями:

В регистре PROGRAM установлен бит read_interleaved.

COMMAND[15:0] = 0x3000.

3. Выполнятся команда Change Read Column Enchanced по схеме «С чтением из FIFO» со следующими особенностями:

В PROGRAM установлен бит read_column_enhanced.

COMMAND[15:0] = 0xE006.

4. Выполнятся команда Change Read Column Enchanced по схеме «С чтением из FIFO» со следующими особенностями:

В PROGRAM установлен бит read_column_enhanced.

COMMAND[15:0] = 0xE006.

29.3.18 Команда Multi-plane Read с ECC

Команда Read Multi-plane (00h-32h) позволяет производить одновременную загрузку нескольких страниц в различные регистры страниц, что позволяет увеличить производительность системы. Для того чтобы выполнить команду чтения над несколькими блоками одновременно, необходимо перед командой Read (00h-30h) добавить одну или несколько команд Read Multi-plane. Выбор блока для считывания данных из флэш-памяти после её готовности осуществляется командой Change Read Column Enchanced (06h-E0h).

Если требуется обновить адрес стобца (адрес байта внутри страницы) без смены LUN, блока или плоскости вместо команды Change Read Column Enhanced может быть использована команда Changre Read Column (05h-E0h).

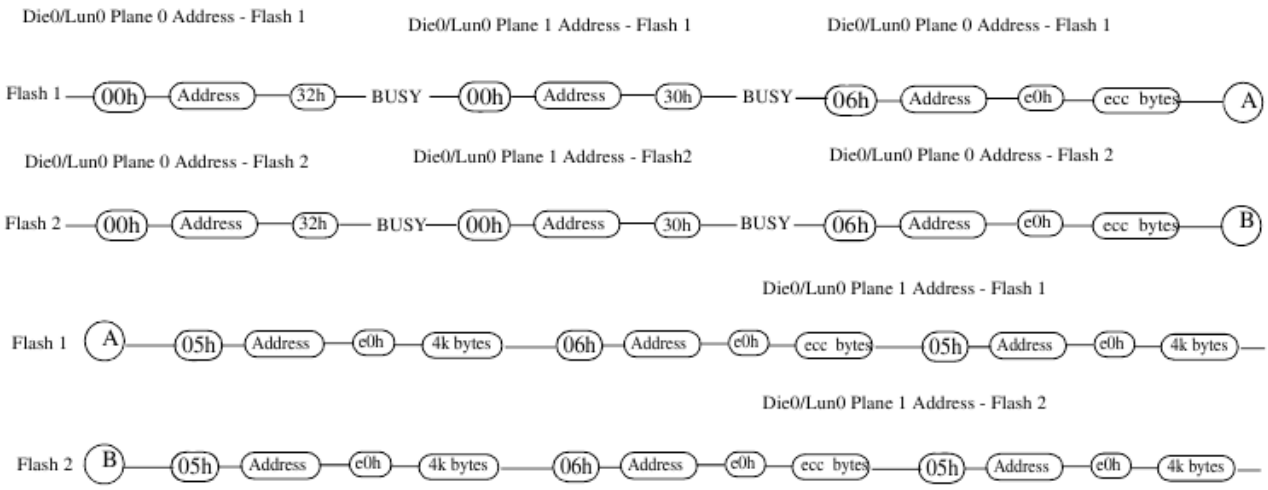


Рисунок 29.12. Одновременное чтение двух блоков памяти из разных флэш-памятей, с ECC для страницы 2k

Параллельное считывание областей из двух разных флэш-памятей (параллельный режим с кодом коррекции ECC:

1. Выполнятся команда Multi-plane Read по схеме «Без доступа к FIFO» со следующими особенностями:

В регистре PROGRAM установлен бит read_interleaved.

COMMAND[15:0] = 0x3200.

2. Выполнятся команда Read по схеме «Без доступа к FIFO» со следующими особенностями:

В регистре PROGRAM установлен бит read_interleaved.

COMMAND[15:0] = 0x3000.

3. Выполнятся команда Change Read Column Enchanced по схеме «С чтением из FIFO» со следующими особенностями:

COMMAND[15:0] = 0xE006, COMMAND[ecc_on_off] = 1

В регистре PROGRAM необходимо установить бит change_read_column_enhanced.

Регистр ECC = 0x0268080C (ecc_addr=16'h080C, ecc_size= 9'h68, ham_bch = 1).

Значение регистра ECC_SPARE_CMD равно 0x2000e005 (ecc_spare_cmd=16’he005, number_of_addr cycles=3’h2).

В результате запуска команды NAND выполнит команду Change Read Column Enhanced и считает байты ECC, а затем выполнит команду Change Read Column и считает байты основных данных.

4. Выполнятся команда Change Read Column Enchanced по схеме «С чтением из FIFO» со следующими особенностями:

COMMAND[15:0] = 0xE006, COMMAND[ECC_ON_OFF] = 1

В регистре PROGRAM необходимо установить бит change_read_column_enhanced.

Регистр ECC = 0x0268080C (ecc_addr=16'h080C, ecc_size= 9'h68, ham_bch = 1).

Значение регистра ECC_SPARE_CMD равно 0x2000e005 (ecc_spare_cmd=16’he005, number_of_addr cycles=3’h2).

В результате запуска команды NAND выполнит команду Change Read Column Enhanced и считает байты ECC, а затем выполнит команду Change Read Column и считает байты основных данных.

29.3.19 Команда Multi-plane Block Erase

Команда Multi-plane Block Erase (60h-D1h) позволяет производить одновременное стирание более чем одного блока в массиве флэш-памяти, что позволяет увеличить производительность системы. Для того чтобы выполнить команду Erase одновременно над несколькими блоками, необходимо перед командой Erase Block (60h-D0h) добавить одну или несколько команд Multi-plane Block Erase.

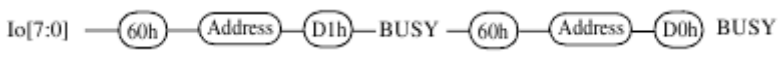


Рисунок 29.13. Одновременное стирание двух блоков памяти в одном LUN

Пример одновременного стирания двух блоков памяти в одном LUN:

1. Выполняется команда Block Erase Multi-plane по схеме «Команды без доступа к FIFO», со следующими особенностями:

В регистре PROGRAM установлен разряд block_erase.

COMMAND[15:0] = 0xD160.

2. Выполнятся команда Block Erase (см. Описание команды Block Erase)

29.3.20 Команда Page Cache Program (2 interleaved адреса)

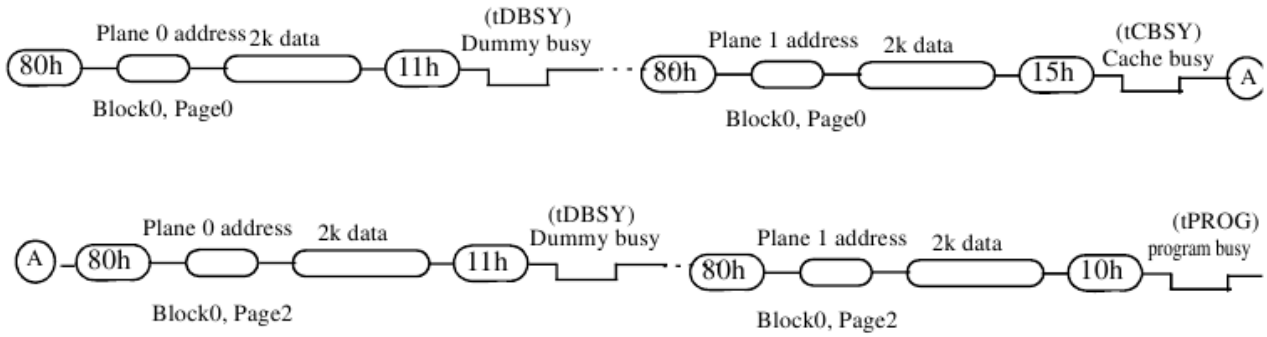


Рисунок 29.14. Page Cache Program (2 interleaved адреса)

Выполнение последовательности:

1. Выполнятся команда Multi-plane Page Program.

2. Выполнятся команда Page Cache Program по схеме «С записью в FIFO» со следующими особенностями:

В регистре PROGRAM установлен бит page_program.

3. Выполнятся команда Multi-plane Page Program.

4. Выполнятся команда Page Program.

29.3.21 Команда Page Cache Program

Команда Page Cache Program (80h-15h) используется для увеличения производительности операций программирования.

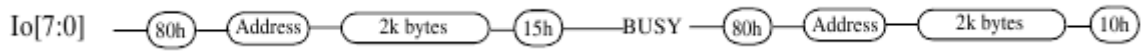


Рисунок 29.15. Page Cache Program

Пример использования:

1. Команда Page Cache Program выполняется по схеме «С записью в FIFO» со следующими особенностями:

COMMAND[15:0] = 0x1580.

В регистре PROGRAM необходимо установить бит page_program.

2. Команда Page Program выполняется по схеме «С записью в FIFO», со следующими особенностями:

COMMAND[15:0] = 0x1080.

В регистре PROGRAM необходимо установить бит page_program.

29.3.22 Команда Read Cache Sequential

Использование серии команд Read Cache Sequential (31h) позволяет увеличить скорость передачи данных. Этого достигается за счёт того, что во момент выдачи данных из регистра кэша, новая (следующая, последовательная) страница данных загружается из массива флэш-памяти в регистр данных.

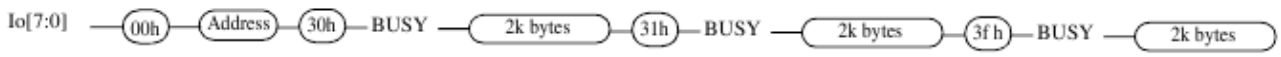


Рисунок 29.16. Read Cache Sequential

Пример использования:

1. Выполняется команда Read по схеме «Без доступа к FIFO» со следующими особенностями:

COMMAND[15:0] = 0x3000.

В регистре PROGRAM необходимо установить бит read_cache_start.

2. Выполняется команда Read Cashe Sequential по схеме «С чтением из FIFO», со следующими особенностями:

COMMAND[7:0] = 0x31.

В PROGRAM необходимо установить бит read_cache_sequential.

3. Выполняется команда Read Cashe End по схеме «С чтением из FIFO», со следующими особенностями:

COMMAND[7:0] = 0x3F.

В PROGRAM необходимо установить бит read_cache_end.

29.3.23 Команда Read Cache Random

Команда Read Cache Random (00h-31h) начинает копирование заданной страницы из массива флэш-памяти в соответствующий регистр данных.

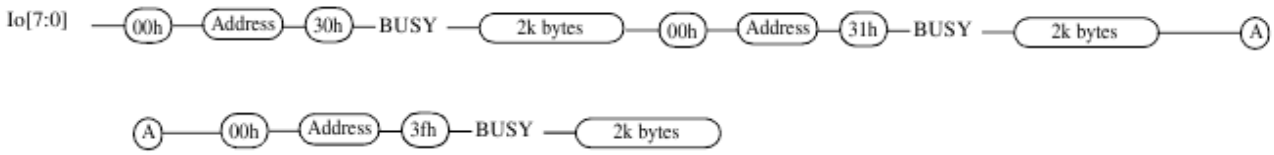


Рисунок 29.17. Read Cache Random

Пример использования:

1. Выполняется команда Read по схеме «Без доступа к FIFO» со следующими особенностями:

COMMAND[15:0] = 0x3000.

В регистре PROGRAM необходимо установить бит read_cache_start.

2. Выполняется команда Read Cashe Random по схеме «С чтением из FIFO», со следующими особенностями:

COMMAND[15:0] = 0x3100.

В PROGRAM необходимо установить бит read_cache_random.

3. Выполняется команда Read Cashe End по схеме «С чтением из FIFO», со следующими особенностями:

COMMAND[7:0] = 0x3F.

В PROGRAM необходимо установить бит read_cache_end.

29.3.24 Выбор LUN/die

LUN – наименьший элемент, который может независимо выполнять команды и сообщать о своём состоянии. Для увеличения производительности при работе с устройствами, которые состоят из нескольких LUN, можно параллельно выполнять операции с разными LUN. NADMPORT поддерживает следующие операции над несколькими LUN:

1. Page Program LUN0, Page Program LUN1.
2. Read LUN0, Read LUN1.
3. Erase LUN0, Erase LUN1.

4. Page Program LUN0, Read LUN1.

Для выбора необходимого из нескольких LUN используется команда Read Status Enhanced. После выполнения этой команды только LUN, который был выбран, будет откликаться на циклы чтения данных.

Последовательность работы с несколькими LUN:

1. Выбор LUN0.
2. Запуска команды для LUN0 (Program, Erase, Read).
3. Установка занятости LUN0.
4. Выбор LUN1 (в этот момент LUN0 занят пересылкой).
5. Запуска команды для LUN1 (Program, Erase, Read).
6. Ожидание завершения операций.

29.3.25 Команда Read Status Enhanced

Данная команда выполняется по схеме «Без доступа к FIFO», со следующими особенностями:

1. Перед выполнением запуска команды через регистр PROGRAM необходимо дополнительно настроить регистр PACKET (packet_size = 1, packet_count = 1). Настройка для режима NV-DDR и NV-DDR2: packet_size = 2.
2. В регистре PROGRAM необходимо установить бит read_status_enhanced.

После выполнения команды процессор может произвести чтение регистр FLASH_STATUS (данные о состоянии последней операции были загружены из флэш-памяти в этот регистр в результате выполнения команды).

29.3.26 Page Program LUN0, Page Program LUN1

Последовательность программирования LUN0, LUN1:

1. Командой Read Status Enhanced выбирается LUN0.
2. Выполняется команда Page Program по схеме «С записью в FIFO» со следующими особенностями:
3. В регистре PROGRAM необходимо установить биты page_program и multi_die.
4. Командой Read Status Enhanced выбирается LUN1.

5. Выполняется команда Page Program по схеме «С записью в FIFO» со следующими особенностями:
6. В регистре PROGRAM необходимо установить бит page_program.

Если необходимо произвести программирование нескольких LUN, бит multi_die совместно с page_program в регистре PROGRAM должен быть установлен для всех команд чтения, кроме последней.

Пример программирования LUN0, LUN1, LUN2:

Для LUN0 и LUN1 бит multi_die необходимо устанавливать совместно с page_program, а для LUN2 только page_program.

29.3.27 Read LUN0, Read LUN1

Последовательность чтения из LUN0 и LUN1:

1. Командой Read Status Enhanced выбирается LUN0.
2. Выполняется команда Read по схеме «Без доступа к FIFO» со следующими особенностями:
3. COMMAND[15:0] = 0x3000.
4. В регистре PROGRAM необходимо установить биты read и multi_die.
5. Командой Read Status Enhanced выбирается LUN1.
6. Выполняется команда Read по схеме «Без доступа к FIFO» со следующими особенностями:
7. COMMAND[15:0] = 0x3000.
8. В регистре PROGRAM необходимо установить биты read и multi_die.
9. Командой Read Status Enhanced выбирается LUN0.
10. Данный шаг повторяется до тех пор, пока в 6-ом (RDY) разряде регистра FLASH_STATUS не будет считана единица (признак того, что LUN готов принимать следующие команды и остальные разряды FLASH_STATUS достоверны).
11. Выполняется команда по схеме «С чтением из FIFO» со следующими особенностями:
12. COMMAND[7:0] = 0x00.
13. В PROGRAM необходимо установить бит 5 (multi_die_rd).

14. Командой Read Status Enhanced выбирается LUN1.

15. Данный шаг повторяется до тех пор, пока в 6-ом (RDY) разряде регистра FLASH_STATUS не будет считана единица (признак того, что LUN готов принимать следующие команды и остальные разряды FLASH_STATUS достоверны).

16. Выполняется команда по схеме «С чтением из FIFO» со следующими особенностями:

17. COMMAND[7:0] = 0x00.

18. В PROGRAM необходимо установить бит 5 (multi_die_rd).

29.3.28 Block Erase LUN0, Block Erase LUN1

Последовательность стирания в LUN0 и LUN1:

1. Командой Read Status Enhanced выбирается LUN0.

2. Выполняется команда по схеме «Без доступа к FIFO» со следующими особенностями:

3. В регистре PROGRAM необходимо установить бит 2 (block_erase) и бит 1 (multi_die).

4. Командой Read Status Enhanced выбирается LUN1.

5. Выполняется команда по схеме «Без доступа к FIFO» со следующими особенностями:

6. В регистре PROGRAM необходимо установить бит 2 (block_erase).

Если требуется выполнить команду Erase для нескольких блоков, бит multi_die совместно с block_erase в регистре PROGRAM должен быть установлен для всех команд, кроме последней.

29.3.29 Page Program LUN0, Read LUN1

Последовательность Page Program в LUN0 и Read в LUN1:

1. Командой Read Status Enhanced выбирается LUN0.

2. Выполняется команда по схеме «С записью в FIFO» со следующими особенностями:

3. В регистре PROGRAM необходимо установить бит 4 (page_program) и бит 1 (multi_die).

4. Командой Read Status Enhanced выбирается LUN1.
5. Выполняется команда по схеме «С чтением из FIFO» со следующими особенностями:
6. В регистре PROGRAM необходимо установить бит 0 (read).

29.3.30 Small Data Move

Если флэш-память поддерживает команду Small Data Move, процессор может записывать данные во флэш-память с приращением меньшим, чем размер страницы (применимо для команд Program и Copyback). Наименьший размер данных, которые можно записать во флэш-память, используя Small Data Move – двойное слово. При выполнении команд Small Data Move поддержка ECC должна быть отключена. Процессор может определить наличие поддержки Small Data Move во флэш-памяти, произведя чтение страницы параметров.

29.3.31 Small Data Move, Page Program

На следующем рисунке показан пример программирования 4-х областей флэш-памяти с размером страницы 2k, использующий свойство Small Data Move:

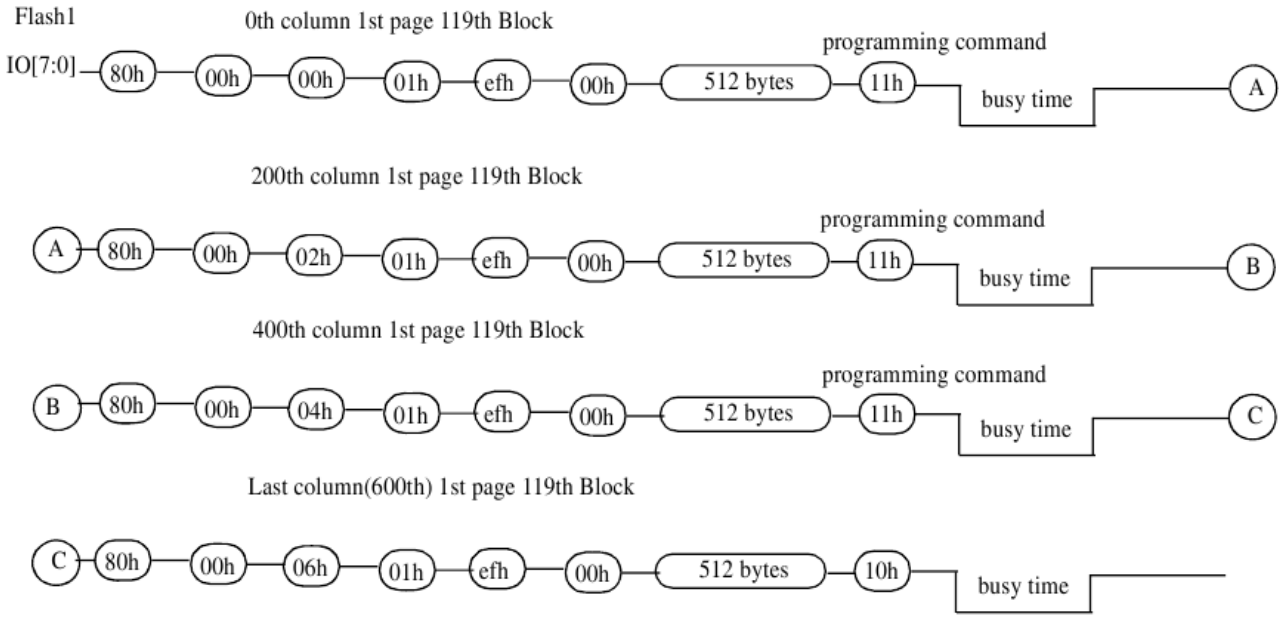


Рисунок 29.18. Пример записи во флэш-память – Small Data Move

При обычном программировании страницы размером 2k достаточно было выполнить одну команду программирования, например, настроив 4 пакета по 512 байт. Small Data Move (с размером пакета 512 байт и количеством пакетов 1) требует выполнения нескольких команд:

1. Multi-plane Page Program по схеме «С записью в FIFO»: COMMAND[15:0] = 0x1180, MEMADDR1 = 000, в PROGRAM установлен small_data_move.

2. Multi-plane Page Program по схеме «С записью в FIFO»:

COMMAND[15:0] = 0x1180, MEMADDR1 = 200, в PROGRAM установлен small_data_move.

3. Multi-plane Page Program по схеме «С записью в FIFO»:

COMMAND[15:0] = 0x1180, MEMADDR1 = 400, в PROGRAM установлен small_data_move.

4. Multi-plane Page Program по схеме «С записью в FIFO»:

COMMAND[15:0] = 0x1080, MEMADDR1 = 600, в PROGRAM установлен small_data_move.

29.3.32 Small Data Move, Copyback Program

Выполняется аналогично Page Program, Small Data Move. Единственное отличие в настройке кода операции в регистре PROGRAM, т.е. 85h вместо 80h.

29.3.33 Small Data Move, Change Read Column Enhanced

После запуска команд Small Data Move процессор может использовать команду Change Read Column Enhanced для чтения данных, записанных до этого:

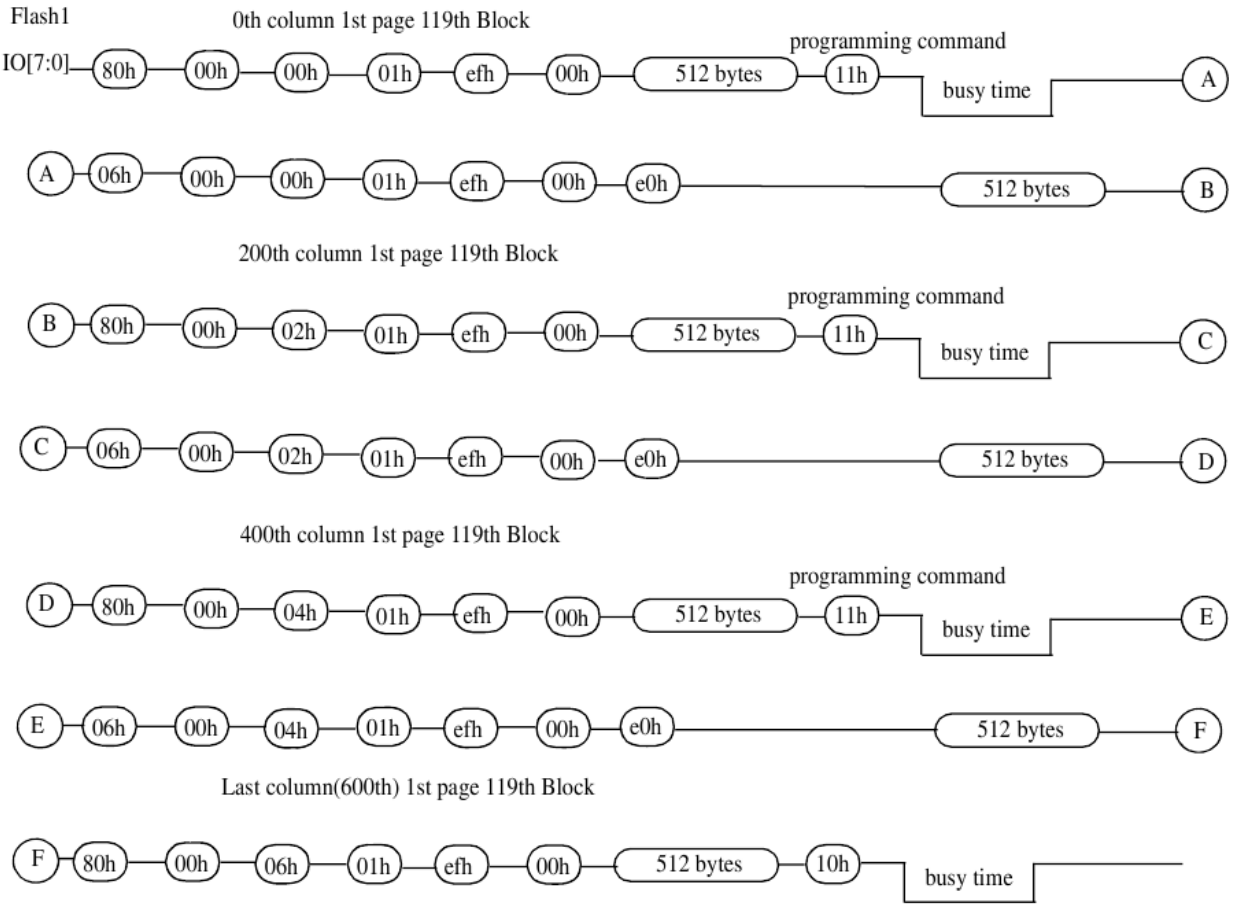


Рисунок 29.19. Small Data Move, Change Read Column Enhanced

Команда Change Read Column Enhanced выполняется по схеме «С чтением из FIFO»: COMMAND[15:0] = 0xE006, в PROGRAM установлен change_read_column_enhanced.

29.3.34 Команда Change Row Address

Команда Change Row Address изменяет адрес строки и столбца LUN, куда производится запись. Можно использовать совместно с командой программирования флэш-памяти (адрес LUN и interleaved адреса должны быть те же, что и в команде программирования):

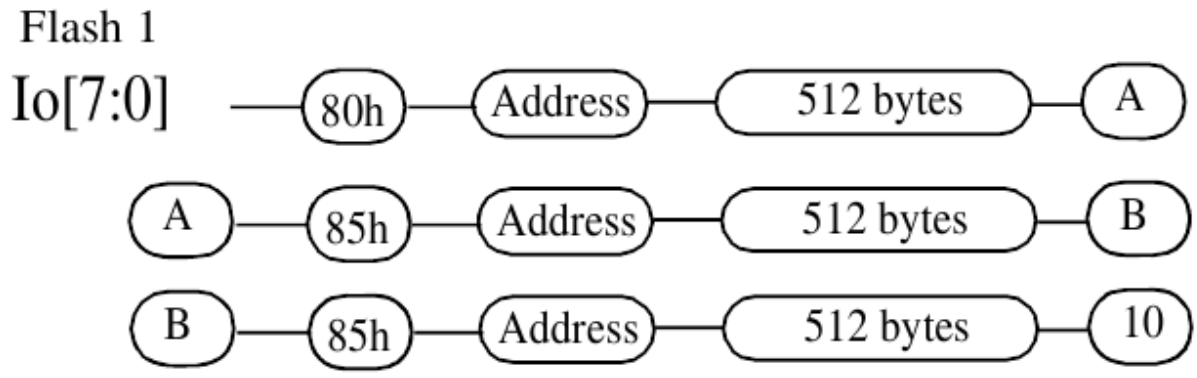


Рисунок 29.20. Change Row Address

В данном примере процессору необходимо запрограммировать следующие области флэш-памяти:

- LUN0, Plane0, Блок 1, Страница 0 (512 байт);
- LUN0, Plane0, Блок 2, Страница 0 (512 байт);
- LUN0, Plane0, Блок 3, Страница 0 (512 байт).

Для этого выполняются команды:

1. Page Program по схеме «С записью в FIFO»:

COMMAND[7:0] = 0x80.

В регистре PACKET поле packet_count = 1 и packet_size = 512.

В PROGRAM установлены change_row_addr и page_program.

2. Change Row Address по схеме «С записью в FIFO»:

COMMAND[7:0] = 0x85.

В регистре PACKET поле packet_count = 1 и packet_size = 512.

В PROGRAM установлен change_row_addr.

3. Change Row Address по схеме «С записью в FIFO»:

COMMAND[15:0] = 0x1085.

В регистре PACKET поле packet_count = 1 и packet_size = 512.

В PROGRAM установлен change_row_addr_end.

Шаг 2 может повторятся несколько раз, если необходимо запрограммировать несколько страниц.

29.3.35 Команда Reset LUN

Команда Reset LUN используется для сброса логического устройства (LUN).

Команда выполняется по схеме «Команды без доступа к FIFO», со следующей особенностью:

в регистре PROGRAM необходимо установить разряд reset_lun.

29.3.36 Свойство Page Register Clear Enhancement

В стандарте ONFI 2.1 (пункт 3.1.3. Multiple LUN Operation Restrictions) указано, что не следует начинать программирование LUN, если выполняется операция чтения другого LUN. Связано с тем, что в ONFI 2.1 при поступлении команды программирования LUN допускается сброс регистров страниц других LUN. И если начать операцию программирования, не завершив операцию чтения, данные чтения, загруженные в регистр страницы, могут быть утеряны.

В стандарте ONFI 2.2 (пункт 3.1.3. Multiple LUN Operation Restrictions) ограничение на запись во время чтения ослаблено. В случае, если флэш-память поддерживает свойство “Page Register Clear Enhancement” (определяется в Странице Параметров) и командой Set Features установлен параметр PC в свойстве “Timing Mode” (см. ONFI 2.2), запись во время чтения допускается.

Для программирования флэш-памяти (команда 80h) во время чтения (для флэш-памяти с установленным признаком PC в свойстве “Timing Mode”) необходимо использовать особое время tADL (время установки данных для программирования после выдачи команды программирования). Время tADL настраивается в регистре TIMING на основании значений байтов 154-155 Страницы Параметров. Для того, чтобы контроллер использовал это время, необходимо установить разряд pgm_pg_reg_clr в регистре PROGRAM.

29.3.37 Команда Set Features

Команда Set Features позволяет изменить значение необходимого параметра флэш-памяти. Команда используется для включения настроек, которые отключены по сбросу питания (смена временного режима, тип интерфейса данных).

Команда выполняется по схеме «С записью в FIFO» со следующей особенностью:

в регистре PROGRAM необходимо установить бит set_feature.

29.3.38 Команда Get Features

Команда Get Features позволяет определить текущее значение настроек флэш-памяти.

Команда выполняется по схеме «С чтением из FIFO» со следующей особенностью:

в регистре PROGRAM необходимо установить бит get_feature.

29.3.39 Переключение временных режимов

Переключение временных режимов выполняется с помощью команд Set Feature и Get Feature. Get Feature используется, чтобы определить текущий режим и тип интерфейса

данных. Set Feature используется, чтобы поменять текущий режим и тип интерфейса данных.

Для перехода в другой временных режимо ПО должно выполнить следующую последовательность:

1. Команда Get Features (не обязательно).
2. Команда Set Features.
3. Команда Get Features.

Типы переходов:

- SDR режим в другой SDR режим
- NV-DDR режим в другой NV-DDR режим
- NV-DDR2 режим в другой NV-DDR2 режим

29.3.39.1 Настройка временного режима SDR

В таблице указаны значения параметров P1-P4, которые необходимо задать команде Set Features (через FIFO при возникновении прерывания «Buffer Write Ready»).

Таблица 29.28. Значения параметров P1-P4 команды Set Features (SDR режим)

Режим	Параметры P1-P4
SDR mode 0	32'h0000_0000
SDR mode 1	32'h0000_0001
SDR mode 2	32'h0000_0002
SDR mode 3	32'h0000_0003
SDR mode 4	32'h0000_0004
SDR mode 5	32'h0000_0005

29.3.39.2 Настройка временного режима NV-DDR

В таблице указаны значения параметров P1-P4, которые необходимо задать команде Set Features (через FIFO при возникновении прерывания «Buffer Write Ready»).

Таблица 29.29. Значения параметров P1-P4 команды Set Features (NV-DDR режим)

Режим	Параметры P1-P4
NV-DDR mode 0	64'h0000_0000_0000_1010
NV-DDR mode 1	64'h0000_0000_0000_1111
NV-DDR mode 2	64'h0000_0000_0000_1212
NV-DDR mode 3	64'h0000_0000_0000_1313
NV-DDR mode 4	64'h0000_0000_0000_1414
NV-DDR mode 5	64'h0000_0000_0000_1515

После успешного завершения команды Set Features процессору следует настроить необходимую частоту clk_flash.

29.3.39.3 Настройка временного режима NV-DDR2

В таблице указаны значения параметров P1-P4, которые необходимо задать команде Set Features (через FIFO при возникновении прерывания «Buffer Write Ready»).

Таблица 29.30. Значения параметров P1-P4 команды Set Features (NV-DDR2 режим)

Режим	Параметры P1-P4
NV-DDR2 mode 0	64'h0000_0000_0000_2020
NV-DDR2 mode 1	64'h0000_0000_0000_2121
NV-DDR2 mode 2	64'h0000_0000_0000_2222
NV-DDR2 mode 3	64'h0000_0000_0000_2323
NV-DDR2 mode 4	64'h0000_0000_0000_2424
NV-DDR2 mode 5	64'h0000_0000_0000_2525

После успешного завершения команды Set Features процессору следует настроить необходимую частоту clk_flash.

29.3.40 Переключение типа интерфейса данных

Возможны следующие переключения:

- SDR – NV-DDR
- SDR – NV-DDR2
- NV-DDR – SDR
- NV-DDR2 – SDR

29.3.40.1 Переключение из SDR в NV-DDR и NV-DDR

Для переключения необходимо выполнить следующую последовательность:

1. Команда Get Features (не обязательно).
2. Команда Set Features (после успешного завершения необходимо перенастроить на нужное значение clk_flash).
    - Отключить clk_flash.
    - Настроить необходимую частоту.
    - Дождаться установки стабильной частоты, записать выбранный режим в регистр PROGRAM
3. Команда Get Features.

Таблица 29.31. Некоторые параметры режимов SDR, NV-DDR, NV-DDR2

Режим	Параметры P1-P4 Set Features	Частота (МГц)	Период (нс)
NV-DDR mode 0	32'h0000_0010	20	50
NV-DDR mode 1	32'h0000_0011	33	30
NV-DDR mode 2	32'h0000_0012	50	20
NV-DDR mode 3	32'h0000_0013	66	15
NV-DDR mode 4	32'h0000_0014	83	12
NV-DDR mode 5	32'h0000_0015	100	10
NV-DDR2 mode 0	32'h0000_0020	33	30
NV-DDR2 mode 1	32'h0000_0021	40	35
NV-DDR2 mode 2	32'h0000_0022	66	15
NV-DDR2 mode 3	32'h0000_0023	83	12
NV-DDR2 mode 4	32'h0000_0024	100	10
NV-DDR2 mode 5	32'h0000_0025	133	7.5
NV-DDR2 mode 6	32'h0000_0026	166	6
NV-DDR2 mode 7	32'h0000_0027	200	5
NV-DDR2 mode 8	32'h0000_0028	267	3.75

29.3.40.2 Переключение из NV-DDR/NV-DDR2 в SDR режим

Процессору необходимо выполнить следующие действия:

1. Выполнить команду Get Features (не обязательно).
2. Установить значение частоты clk_flash в 100 МГц.
3. Выполнить команду Reset(8’hFF) в режиме SDR mode 0. Флэш-память в любом режиме должна распознать команду сброса, выполненную в SDR mode 0.
4. Выполнить команду Get Features.
5. Выполнить команду Set Features и выбрать необходимый режим.

29.3.41 Page Program (режим SDMA)

В данной версии NAND'A режим SDMA не поддерживается.

29.3.42 Read (режим SDMA)

В данной версии контроллера режим SDMA не поддерживается.

29.3.43 Page Program (режим MDMA)

Команда Page Program пересылает страницу данных (или часть страницы) в регистр страницы. Затем содержимое регистра страницы записывается в массив флэш-памяти. В режиме MDMA данные страницы напрямую загружаются из памяти, используя DMA контроллер.

Схема выполнения команды Page Program в режиме MDMA:

1. CPU разрешает формирование признака прерывания “Transfer Complete” (см. регистр INTERRUPT_STATUS_EN).
2. CPU разрешает формирование сигнала прерывания “Transfer Complete” (см. регистр INTERRUPT_SIGNAL_EN).
3. CPU включает режим MDMA в регистре COMMAND.
4. CPU настраивает регистр PACKET.
5. CPU настраивает регистр DMA_SYS_ADDR.
6. CPU настраивает регистр DMA_BUFFER_BOUNDARY.
7. CPU настраивает регистр MEMADDR1.
8. CPU настраивает регистр MEMADDR2.
9. CPU инициирует выполнение команды установкой разряда page_program в регистре PROGRAM.
10. DMA контроллер считывает данные из системной памяти, сохраняет в FIFO и отправляет во флэш-память.
11. По приходу прерывания от DMA контроллера CPU обновляет значение регистра DMA_SYS_ADDR.
12. После завершения выполнения команды NAND устанавливает признак прерывания “Transfer Complete” в регистре INTERRUPT_STATUS.
13. Формируется физический сигнал прерывания.
14. CPU считывает регистр INTERRUPT_STATUS.
15. CPU сбрасывает бит “Transfer Complete” регистра INTERRUPT_SIGNAL_EN.
16. CPU сбрасывает бит “Transfer Complete” регистра INTERRUPT_STATUS_EN.
17. CPU сбрасывает бит “Transfer Complete” в регистре INTERRUPT_STATUS.

29.3.44 Read (режим MDMA)

Команда Read считывает страницу данных, определяемую адресом строки LUN. Страница данных становится доступной для чтения с заданного адреса столбца. В режиме MDMA данные страницы загружаются в системную память, используя DMA контроллер.

Команда Read выполняется аналогично Page Program в режиме MDMA:

Пункты 1-8 совпадают с командой Page Program (MDMA).

Пункт 9: CPU инициирует выполнение команды установкой разряда read в регистре PROGRAM.

Пункт 10: DMA контроллер считывает данные из FIFO и записывает их в системную память.

Пункты 11-17 совпадают с командой Page Program (MDMA).

29.4 Обнаружение и инициализация флэш-памяти

В одном корпусе флэш-памяти может содержаться от одного до двух независимых компонентов (целевых устройств). Выбор необходимого компонента осуществляется отдельным сигналом CE#.

Обнаружение и инициализация устройств подключённых к CE#

После сброса по включению питания необходимо:

- выполнить команду Reset для целевого устройства;
- определить состояние командой Read Status;
- выполнить команду Read ID с адресом 20h;
- убедиться, что данные, считанные командой Read ID, соответствуют стандарту ONFI. В случае несоответствия стандарту, наличия ошибки/истечения времени ожидания подключать и в дальнейшем использовать проверяемый CE# не следует.

Выполнить команду Read Parameter Page. Данная команда считывает информацию о возможностях и параметрах флэш-памяти. После считывания данных необходимо проверить CRC для подтверждения того, что данные были получены корректно и без ошибок. Если проверка CRC для первой операции Read Parameter Page не прошла, то необходимо считать резервные копии страницы параметров. После успешного извлечения данных из страницы параметров процессор обладает всей необходимой информацией для успешного взаимодействия с целевым устройством.

29.5 Настройка тактирования NAND

Для работы NAND'a необходимо настроить следующие частоты:

- clk_flash – рабочая частота Nand Flash Интерфейса;
- clk_sys – системная частота.

В режиме SDR частота clk_flash должна быть равна 100 МГц.

Для режимов NV-DDR и NV-DDR2 частота clk_flash должна соответствовать выбранному временному режиму. Если командой Set Features был выбран режим N, то рабочая частота должна быть больше частоты режима N-1 и не должна превышать частоту режима N. Например, для временного режима 2 (частота – 50 МГц) должно выполняться следующее соотношение: 33 МГц < clk_flash <= 50 МГц

29.6 Обработка ошибок

Процедура обнаружения и коррекции ошибок применима только к командам программирования или чтения флэш-памяти. В режиме обнаружения и коррекции ошибок при программировании флэш-памяти помимо данных во флэш-память дополнительно записывается контрольная информация. Контрольная информация позволяет обнаруживать и восстанавливать ошибки при последующем чтении (ошибки в данных и в контрольных разрядах).

29.6.1 HAM

При выполнении операции чтения (с использованием коррекции) NAND сравнивает контрольные байты из флэш-памяти с вычисленными на основе считанных данных.

29.6.1.1 Корректируемые ошибки

При обнаружении ошибки в одном бите NAND автоматически исправляет ошибку, и устанавливает прерывание “Single Bit Error”. После приёма пакета устанавливается прерывание “Buffer Read Ready”. По окончанию всей пересылки устанавливается прерывание “Transfer Complete”.

Рассмотрим, следующий пример (размер страницы − 2k, размер пакета − 512, количество пакетов – 4). При считывании первого пакета NAND обнаруживает однобитную ошибку. Затем NAND исправляет её и устанавливает прерывание “Single Bit Error”. После этого контроллер считывает второй, третий и четвертый пакеты из флэш-памяти. Прерывание “Buffer Read Ready” устанавливается после приёма каждого пакета. После чтения всех данных из флэш-памяти NAND устанавливает прерывание “Transfer Complete”.

29.6.1.2 Некорректируемые ошибки

Если контроллер обнаруживает ошибки в двух битах, устанавливается прерывание “Multi Bit Error”. В этом случае прерывания “Buffer Read Ready” и “Transfer Complete” не устанавливаются. Например, размер страницы − 2k, размер пакета – 512 Бит, количество пакетов − 4. При чтении первого пакета данных, если контроллер обнаруживает ошибку в двух битах, устанавливается прерывание “Multi Bit Error”. NAND переключается в состояние IDLE.

29.6.2 BCH

Контроллер позволяет работать в следующих режимах с коррекцией в кодах БЧХ:

- BCH-4. Коррекция и обнаружение не более 4 ошибок на каждые 512 байт;
- BCH-8. Коррекция и обнаружение не более 8 ошибок на каждые 512 байт;
- BCH-12. Коррекция и обнаружение не более 12 ошибок на каждые 512 байт;
- BCH-16. Коррекция и обнаружение не более 16 ошибок на каждые 512 байт;
- BCH-24. Коррекция и обнаружение не более 24 ошибок на каждые 1024 байта.

В зависимости от размера программируемой или считываемой из флэш-памяти страницы пересылка разбивается на необходимое количество пакетов размером 512 или 1024 байта. Для корректной работы в режимах BCH-4, BCH-8, BCH-12, BCH-16 поле packet_size регистра PACKET должно быть установлено в значение 512. Для корректной работы в режиме BCH-24 поле packet_size регистра PACKET должно быть установлено в значение 1024. Пример настройки PACKET см. в Разделе «Регистр PACKET».

Последовательность программирования и чтения флэш-памяти с использованием кодов коррекции БЧХ:

1. Настройка контроллера и запуск пересылки (см. описание команд Program).
2. Контроллер записывает данные всех пакетов последовательно в основную область страницы.
3. После пересылки данных всех пакетов в конец spare-область страницы записываются коды коррекции, вычисленные на основе записанных данных.
4. Настройка контроллера и запуск пересылки (см. описание команд Read).
5. Считываются коды коррекции из spare-области страницы.
6. После считывания кодов коррекции считываются данные всех пакетов, и осуществляется контроль их соответствия считанным кодам. Если при считывании пакета была обнаружена ошибка, то будет установлено прерывание «BCH Detect Error» (если разрешено). Прерывание сформируется по окончанию приёма пакета при обнаружении хотя бы одной ошибки независимо от того, была ли она исправлена или нет.

29.6.3 Чтение стёртых страниц с ECC

В следующей таблице описаны особенности считывания стёртых страниц флэш-памяти (данные и spare-область заполнены значением 0xFF) с включенной коррекцией ошибок при bch_residue_xor = 0 (см. регистр COMMAND):

Таблица 29.32. Особенности чтения стёртых страниц с ECC при bch_residue_xor = 0

Режим ECC \Размер страинцы	512	2048	4096	8192
HAM	TBD	TBD	FF	FF
BCH-4	TBD	не FF	Не FF	Не FF
BCH-8	TBD	FF	FF	FF
BCH-12	X	TBD	Не FF	FF
BCH-16	X	TBD	FF	FF
BCH-24	X	TBD	Не FF	Не FF

X – режим недоступен при данном размере страницы
TBD – требует уточнения
FF – при чтении стёртой страницы возвращаются множественные значения 0xFF
Не FF – при чтении стёртой страницы возвращаются значения отличные от 0xFF.

Если бит bch_residue_xor регистра COMMAND установлен в 1, то чтение стёртой страницы не вызывает ошибки, и ничем не отличается от чтения страницы, которая была прописана значениями 0xFF.. В этом режиме код коррекции модифицируется таким образом, что значение 0xFF.. считается валидным значением.

29.7 Прерывания

Работа с прерываниями осуществляется через регистры INTERRUPT_STATUS_EN, INTERRUPT_SIGNAL_EN, INTERRUPT_STATUS.

В регистре INTERRUPT_STATUS_EN задаётся разрешение установки необходимого признака. В регистре INTERRUPT_STATUS отображается текущее состояние признаков прерываний. Регистр INTERRUPT_SIGNAL_EN разрешает или запрещает формирование физического сигнала прерывания.

Возможно формирование прерывания по следующим событиям (см. описание регистра INTERRUPT_STATUS):

- «Buffer Write Ready»;
- «Buffer Read Ready»;
- «Transfer Complete»;
- «Multi Bit Error»;
- «Single Bit Error»;
- «dma_int»;
- «error_sys».

30. КОНТРОЛЛЕР QUAD SPI (QSPI)

30.1 Общая информация

Контроллер QSPI предназначен для подключения SPI устройств.

Основные параметры:

- Поддержка чтения флэш-памятей крупнейших производителей в режиме XIP
- Аппаратный интерфейс к контроллеру DMA
- Программное переключение между режимами мастера и слейва
- Программируемая скорость SCLK в режиме мастер
- Поддержка режимов:
    - 4-проводной режим (Quad)
    - 2-проводной режим (Dual)
    - Полнодуплексный
    - Полудуплексный
- Поддержка форматов SPI:
    - Motorola SPI
    - Синхронный последовательный кадр TI
    - Кадр National Microwire
- LSB или MSB режимы
- Подключение до 4 slave-устройств
- Прерывания по таймауту приёма данных
- Прерывание при переполнении FIFO приёма

30.2 Функции

- Интерфейс AXI AMBA
- Функционал исполнения на месте (XIP) под несколько FLASH устройств общепринятого стандарта
- Интерфейс DMA
- Программно задаваемый Master или Slave режим.
- Программно задаваемый тактовый сигнал SCLK в Master режиме.
- Четырёх- и двухбитный режим работы.
- Последовательный приём и передача битовых слов от 4 до 32 бит с инкрементом в 1 бит.
- Полнодуплексный режим
- Поддержка полудуплексного режима
- Отдельный вход тактового сигнала SCLK в Master режиме.
- Буферы FIFO для приёма и передачи данных от 8 до 256 слов, конфигурируемые.
- Асинхронный Slave-интерфейс
- Управление прерываниями
- Режим младшего или старшего бита (LSB или MSB)
- Возможность контролировать до 4 slave-устройств через одно master-устройство.
- Тристабильный сигнальный протокол MISO (много вводов/один вывод) для двух и более slave-устройств.
- Совместимость со множеством общепринятых FLASH устройств.
- Дополнительные GPOs/GPIs для поддержки нестандартных потребностей передачи сигнала.
- Поддержка формата последовательного периферийного интерфейса (SPI) Motorola
- Поддержка формата последовательного синхронного кадра Texas Instruments.
- Поддержка формата кадра National Microwire.
- Режим внешнего подключения ввода
- Прерывание таймаута приёма данных
- Прерывание переполнения буфера FIFO приёма

30.3 Основное назначение

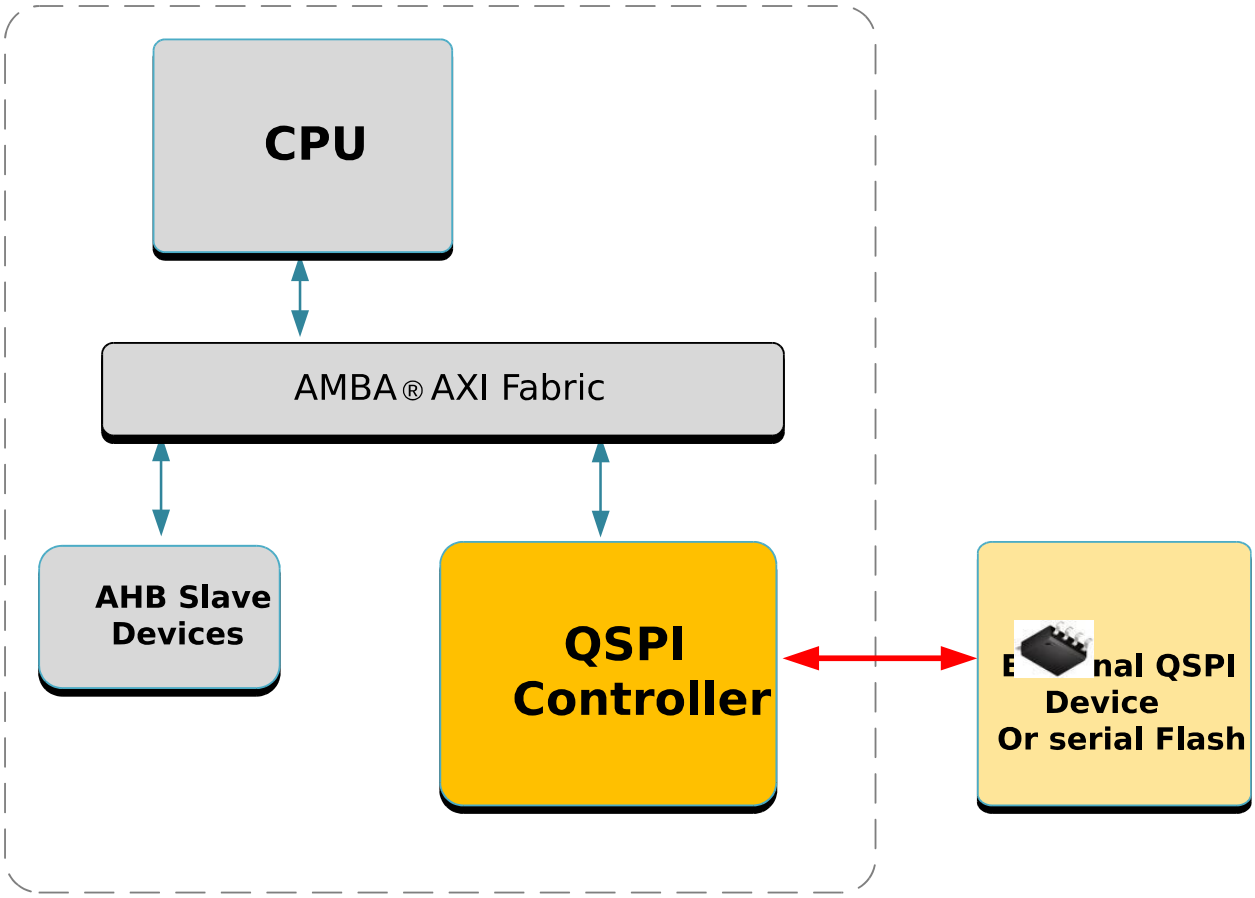


Рисунок 30.1. Основное назначение

30.4 Обзор

Модуль четырехбитного последовательного периферийного интерфейса либо управляет каналом последовательной передачи данных как master-устройство, либо отвечает ему как slave-устройство.

Контроллер шины IPC-QSPI-AXI может быть сконфигурирован программно как master- или как slave-устройство. Чтение и запись в ядро осуществляется через интерфейс шины AHB AMBA. Ядро работает в различных режимах обработки данных, от 4-битного и вплоть до 32-битного.

Затем данные упорядочиваются и передаются (начиная либо с младшего, либо со старшего бита) через стандартный четырёхпроводной интерфейс шины SPI или шины расширенного Quad-режима (четырехбитного режима) SPI.

30.5 Блок-схема

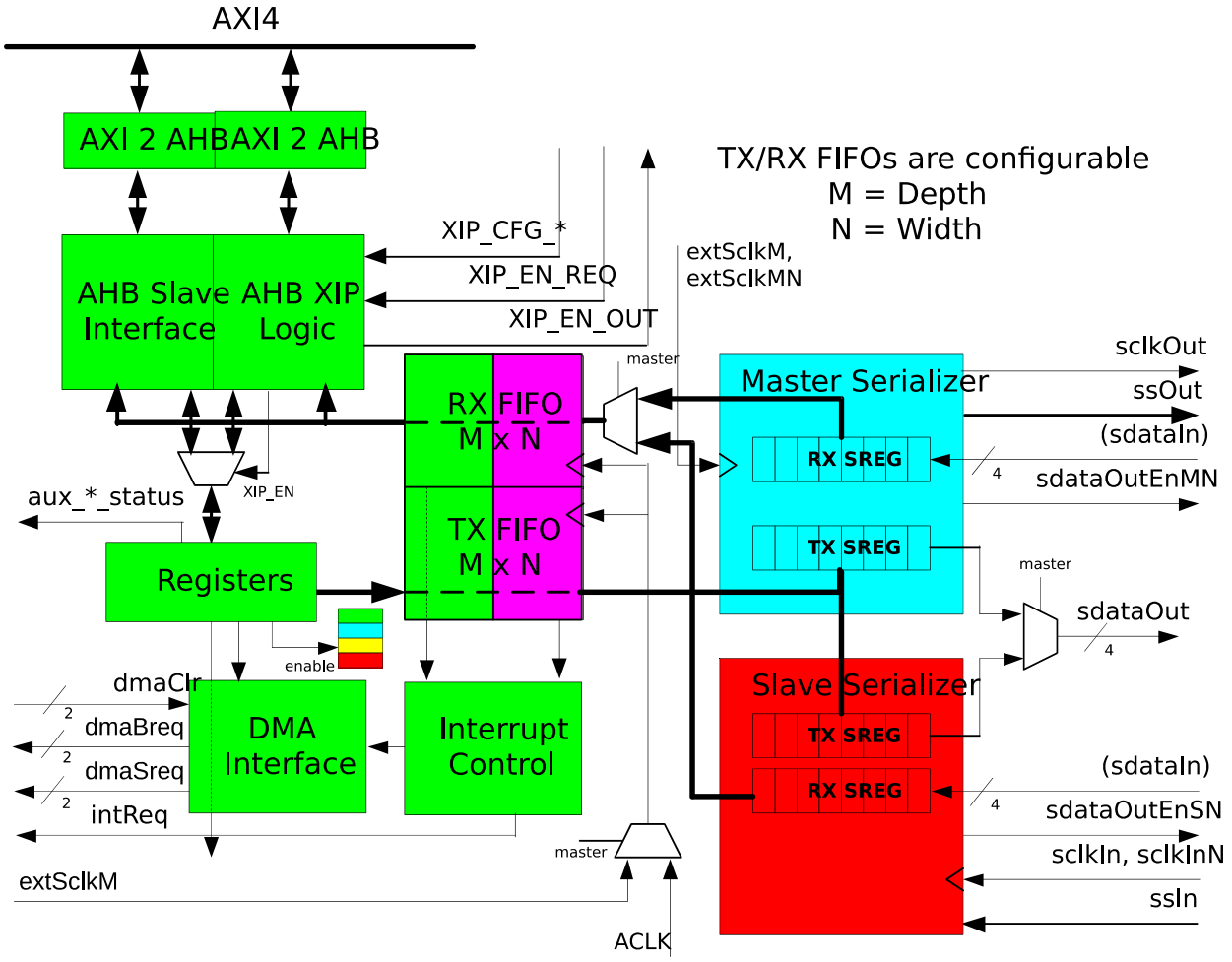


Рисунок 30.2. Блок-схема IPC-QSPI-AXI

На блок-схеме отображены несколько тактовых доменов: тактовый сигнал AXI (выделен зеленым), SCLK внешнего Master-устройства (выделен голубым) и SCLK Slave-устройства (выделен красным). Буферы FIFO служат границей между тактовым сигналом AXI и тактовым сигналом шины SPI. Источник тактового сигнала шины SPI меняется в зависимости от режима Master/Slave; в Master-режиме, ядро QSPI генерирует тактовый сигнал шины SPI напрямую из SCLK внешнего Master-устройства, а в Slave-режиме ядро QSPI получает тактовый сигнал шины SPI от другого Master-устройства SPI.

В Master-режиме, сериализатор Master-устройства и буфера FIFO справа на рисунке (выделены розовым) делят один и тот же тактовый сигнал.

В Slave-режиме, сериализатор Slave-устройства (выделен красным) тактируется напрямую кусочным тактовым сигналом шины SPI sclkIn, а буфера FIFO справа на рисунке (выделены розовым) тактируются сигналом HCLK.

В Slave-режиме либо ssIn, либо внутренний сигнал, который сравнивает счётчик приёма данных с окончательным значением, синхронизируется с доменом HCLK; указатели

буферов FIFO при этом расширяются на передний фронт этого синхронизированного сигнала с активным высоким уровнем.

Старший бит отвечает за управление большей частью внутреннего мультиплексирования.

Есть и другой очень важный сигнал: сигнал подключения.

Бит подключения синхронизируется с различными тактовыми доменами (при необходимости). Эти синхронизированные сигналы подключения вместе с правильным программным порядком регистров AXI, позволяют логике других доменов безопасно брать пробы (сэмплировать) значений остальных регистров и управляющих битов AXI, поскольку они будут гарантированно стабильны после того как бит подключения будет активирован. Вкратце, бит подключения служит основным синхронизатором всего модуля.

Следует отметить, что существуют раздельные сигналы AXI - для стандартного интерфейса/интерфейса регистров, и для интерфейса режима XIP (исполнения на месте).

Это сделано потому, что интерфейс XIP может иметь куда большую зону охвата при декодировании системных адресов.

30.6 Функциональное описание

30.6.1 Канал данных SPI

Модуль четверного последовательного периферийного интерфейса либо управляет каналом последовательной передачи данных как master-устройство, либо отвечает ему как slave-устройство.

Контроллер шины IPC-QSPI-AXI может быть сконфигурирован программно как master- или как slave-устройство. Чтение и запись в ядро осуществляется через интерфейс шины AXI AMBA. Ядро работает в различных режимах обработки данных, от 4-битного и вплоть до 32-битного (с инкрементом 1 бит).

Затем данные упорядочиваются и передаются (начиная либо с младшего, либо со старшего бита) через стандартный четырёхпроводной интерфейс шины SPI или шину расширенного Quad режима.

Данные передаются синхронно по протоколу MOSI (вывод Master-устройства, ввод Slave-устройства) в соответствии с тактовым сигналом SCLK, генерируемым master-устройством.

Master-устройство также в полнодуплексном режиме получает данные по сигналу MISO (ввод Master-устройства, вывод Slave-устройства).

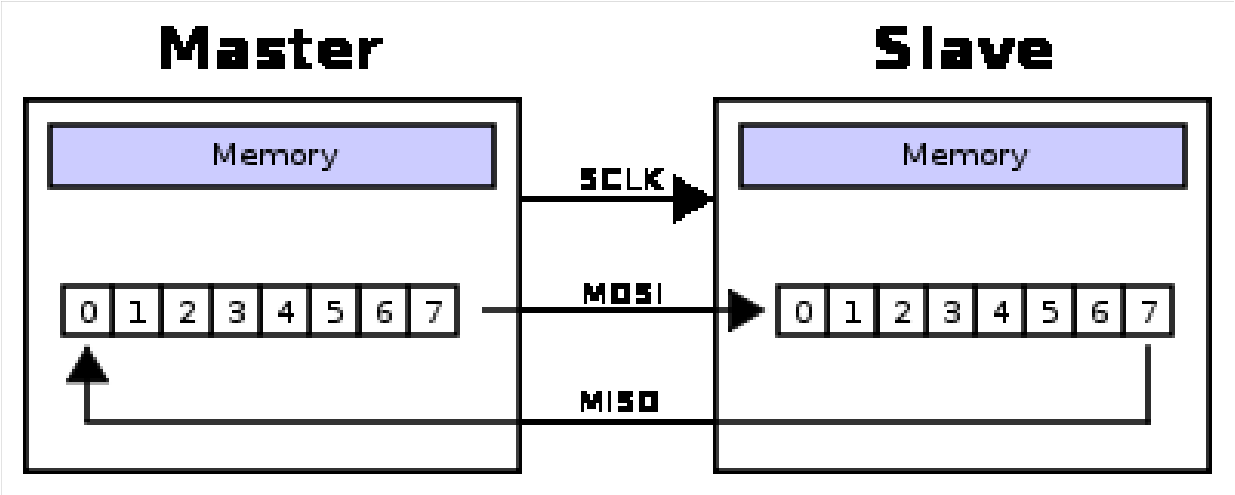


Рисунок 30.3. Передача данных между Master и Slave-устройствами

В Quad-режиме, четыре бита являются одновременно выводом и вводом, и старший бит - всегда sdataOut[3]/sdataIn[3]. В Dual (двухбитном) режиме, два бита являются одновременно выводом и вводом, и старший бит – всегда sdataOut[1]/sdataIn[1]. В однобитных операциях используются только sdataOut[0] и sdataIn[0].

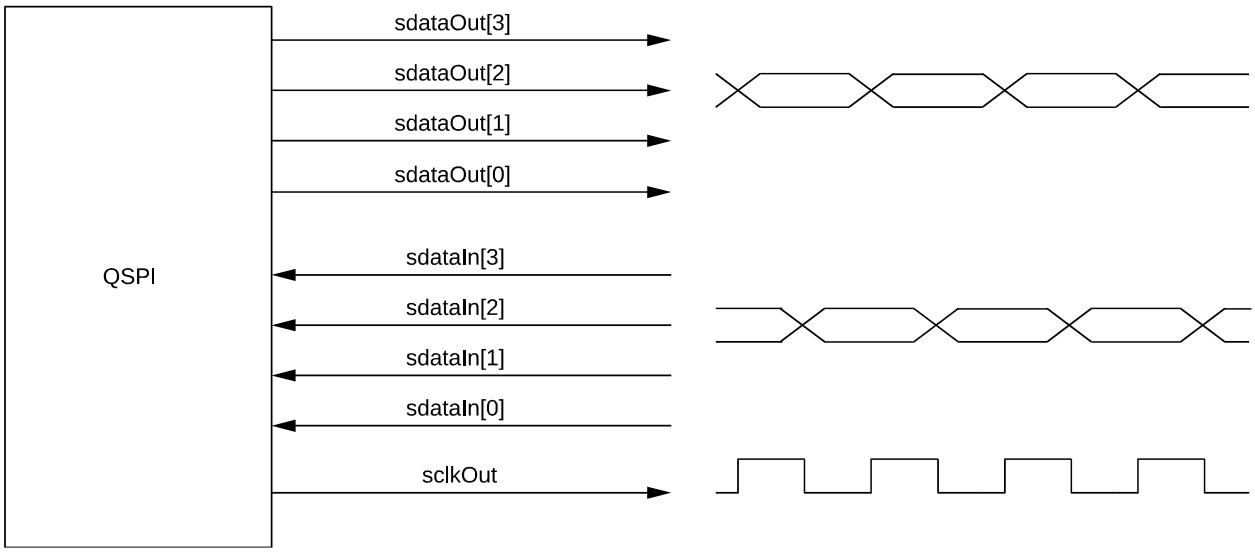


Рисунок 30.4. Master-устройство – Quad SPI

30.6.2 Конфигурация буферов FIFO приема и передачи

Ширина слов буферов FIFO для передачи и приёма IPC-QSPI-AXI может конфигурироваться на стадии компиляции. Разрешенная ширина – степени двойки в диапазоне [8,256]; главным образом, разрешенными являются 8,16,32,64,128,256. Ширины слов FIFO могут быть заданы установкой или блокировкой параметра “QSPI_FIFO_ADDR_BITS” в модуле socAxiQSpi.v.

30.6.3 Работа DMA

Модуль IPC-QSPI-AXI совместим с различными общепринятыми контроллерами DMA. Режим DMA в модуле IPC-QSPI-AXI может быть подключен для помощи контроллеру DMA в загрузке (записи в) буфер FIFO передач и разгрузке (чтению из) буфера FIFO приёма. Для загрузки в буфер передачи, используются биты dmaBreq[0] и dmaClr[0] (dmaSreq[0] не используется). Для разгрузки буфера приема, используются dmaBreq[1], dmaClr[1], и dmaSreq[1]. Раздельные каналы контроллера DMA могут использоваться для управления этими двумя событиями.

Во всех случаях, контроллер DMA действует как регулятор потока.

Загрузка буфера передач может начинаться, когда буфер передач пуст до или ниже своего предельного уровня (watermark level). Предельный уровень TX программируем и устанавливается через регистр управления. Контроллер DMA является тем устройством, которое отслеживает, чтобы записи пакета передачи не превышали объём буфера передачи.

Разгрузка буфера приема может начинаться, когда буфер приема заполнен до или выше своего предельного уровня (watermark level) ИЛИ когда буфер приёма не пуст. Предельный уровень RX программируем и устанавливается через регистр управления.

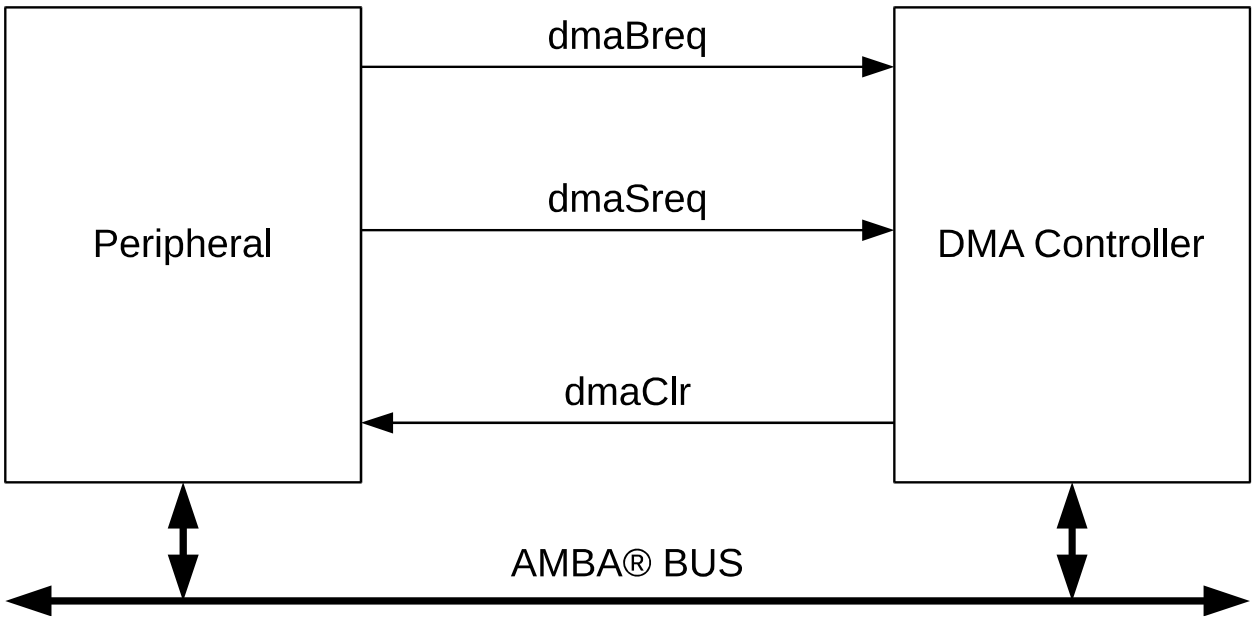


Рисунок 30.5. Работа DMA

30.6.4 Коммутация Master/Slave устройств

Контроллер IPC-QSPI-AXI, сконфигурированный как master-устройство SPI, может быть применён вместе с вплоть до 4-ех slave-устройств SPI. Когда ядро сконфигурировано как slave-устройство, сигнал протокола MISO разделяется на три состояния, чтобы позволить

многим slave-устройствам передавать данные к master-устройству, когда это slave-устройство выбрано.

На рисунке ниже показано взаимодействие устройств в режиме master/slave, где master инициирует передачу кадра данных.

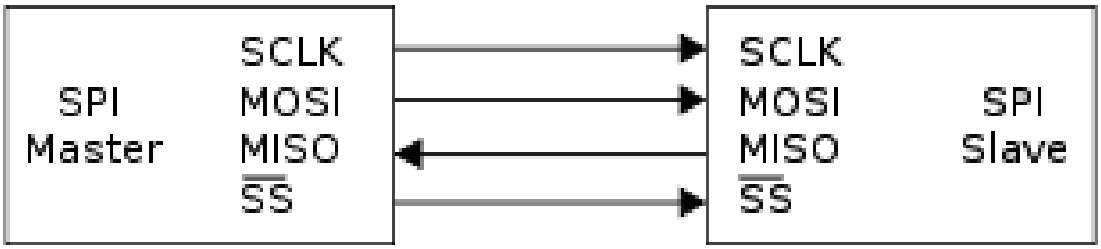


Рисунок 30.6. Коммутация Master - Slave устройств

На рисунке ниже показаны три slave-устройства, подключенные к одному master-устройству по протоколу MISO в виде шины трех состояний.

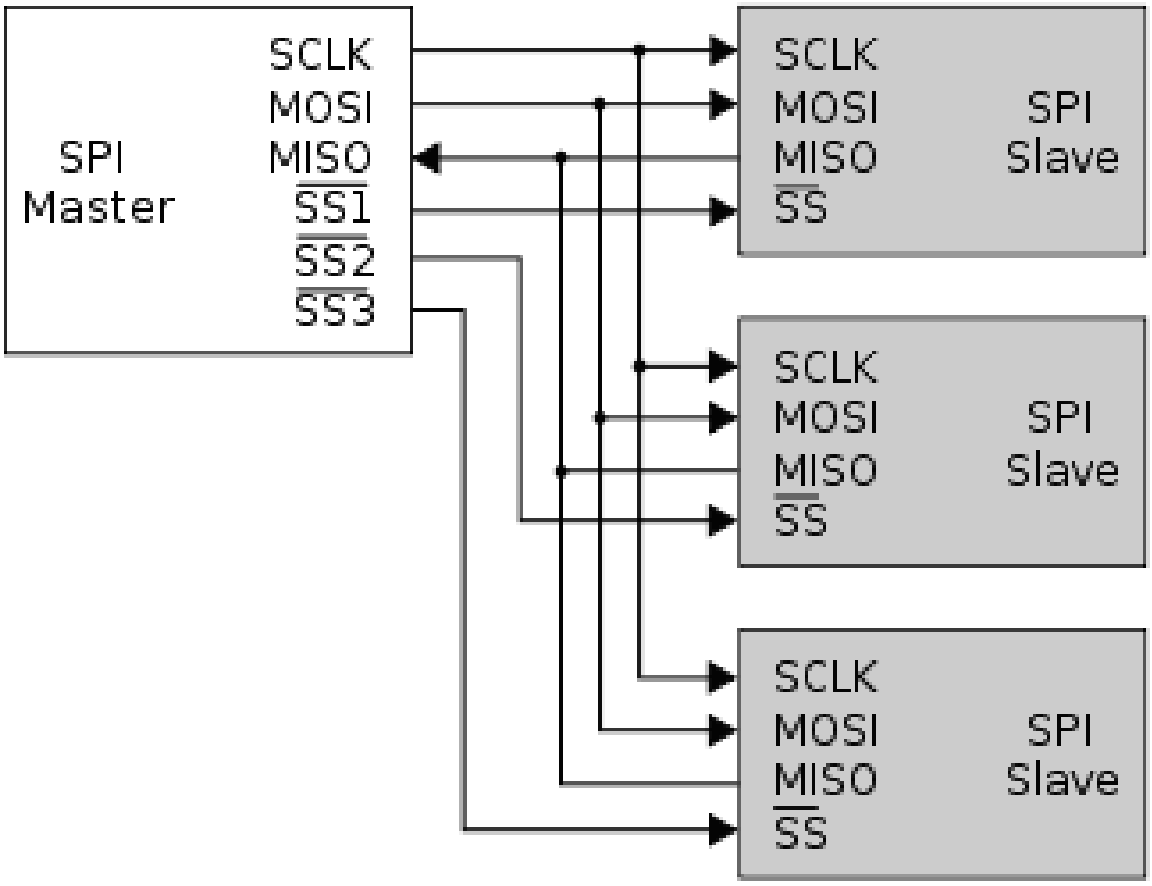


Рисунок 30.7. Коммутация Master-устройства и трёх Slave-устройств

30.6.5 Исполнение на месте (XIP)

Замечание: исполнение на месте (XIP) поддерживается только при FIFO_WIDTH = 32.

Режим XIP – режим, который позволяет Master-устройству AXI напрямую считывать содержимое любого из стандартных устройств с FLASH-памятью просто путём чтения из пространства адресов контроллера QSPI. Это может быть полезно во многих ситуациях, например, для процессора, исполняющего инструкции загрузки с FLASH-устройства QSPI, или для функции копирования DMA, когда контроллер DMA прочитывает содержимое FLASH-устройства QSPI, затем записывает данные в RAM где-либо ещё в системе. Режим XIP следует задействовать только если контроллер QSPI работает как master-устройство с одним из стандартных поддерживаемых семейств FLASH-памяти QSPI, приведённых в таблице ниже.

Таблица 30.1. Семейства поддерживаемых памятей

Изготовитель	Семейство FLASH
Winbond	W25Q
Macronix	MX25L
Spansion	S25FL
Micron	N25Q

В режиме XIP контроллер QSPI функционирует как промежуточный преобразователь (мост) от AXI к QSPI, переводя обращения на чтения AXI в одну из 6 поддерживаемых команд чтения SPI/Dual SPI/Quad SPI для FLASH-памяти. В таблице ниже перечислены поддерживаемые команды чтения и вариации.

Таблица 30.2. Команды чтения в режиме XIP

Название команды	Код команды	Поддержка высокопроизводительного режима
Read	0x03	Нет
Fast Read	0x0B	Нет
Fast Read Dual Out	0x3B	Нет
Fast Read Quad Out	0x6B	Нет
Fast Read Dual I/O	0xBB	Есть
Fast Read Quad I/O	0xEB	Есть

В режиме XIP машина состояний реализует запрошенную команду чтения через управление набором регистров и буферов FIFO контроллера QSPI. При подключении машина состояний XIP начнёт считывать содержимое устройства с FLASH-памятью, начиная с адреса 0. В отсутствие любых обращений на чтения от AXI, машина состояний продолжит считывать содержимое последующих FLASH-адресов кусочками по 32-бита, до тех пор пока буфер RX FIFO не заполнится наполовину.

В любой момент, когда обнаружено обращение на чтение от AXI, выполняется один из следующих сценариев:

1. Запрошенные данные уже содержатся в буфере RX FIFO (или последующие данные в пакете чтения QSPI). Такое происходит если верно логическое И для адреса AXI, большего чем или эквивалентного предыдущему адресу, и адреса AXI меньшего чем или эквивалентного предыдущему адресу + R, где R = 4 x (уровень заполнения буфера RX FIFO + 1).

В этом случае, пакет чтений QSPI продолжает выполняться, и буфер RX FIFO опустошается до тех пор, пока не будут достигнуты запрошенные данные. Затем данные предоставляются запрашивающему Master-устройству AXI.

2. Запрошенные данные находятся за пределами уже содержащихся в RX FIFO. Это происходит если адрес AXI меньше предыдущего адреса, ИЛИ если адрес AXI больше предыдущего адреса + R, R = 4 x (уровень заполнения RX FIFO + 1). В этом случае, RX FIFO очищается, пакетный импульс QSPI на чтение прекращается и инициируется новый пакетный импульс на чтение, начиная с текущего адреса AXI. Данные предоставляются запрашивающему Master-устройству AXI как только становятся доступны в RX FIFO.

В каждом из этих случаев может понадобиться установка большого числа состояний ожидания на шине AXI до тех пор, пока верные данные чтения не будут собраны в RX FIFO и представлены запрашивающему Master-устройству AXI. Это может спровоцировать проблемы с производительностью для стандартной системы AXI, в которой Master-устройство получает доступ к шине AXI через арбитра, используя HBUSREQ/HGRANT.

Однако, для систем, использующих Crossbar-стиль AXI Fabric или для систем, в которых процессор загружается из FLASH-памяти QSPI, производительность вызывает меньше проблем. Так происходит потому, что в этих случаях, другие (не запрашивающие) Master-устройства AXI не будут затронуты всеми состояниями ожидания, запущенными контроллером QSPI.

Записи AXI в FLASH-устройство в XIP-режиме не реализованы; если Master-устройство AXI предпринимает попытку записи в контроллер QSPI пока находится в режиме XIP, запись игнорируется и Master-устройству AXI возвращается ошибка AXI.

30.6.6 Поддерживаемые соотношения тактовых сигналов

Существует несколько методических указаний по относительным частотам тактового сигнала шины SPI и HCLK.

Как в Master, так и в Slave-режиме, тактовым сигналом шины SPI может являться любой сигнал с частотой меньшей или равной частоте HCLK.

Как в Master, так и в Slave-режиме, возможно, что тактовый сигнал шины SPI (extSclkM или sclkIn) с частотой большей, чем частота HCLK будет работать правильно в большинстве случаев, но в целом это не гарантированно.

Аналогичным образом могут появляться ошибки в работе Master-устройства, когда передача SPI занимает менее одного цикла HCLK.

Для режима синхронного последовательного кадра TI, тактовый сигнал шины SPI должен быть больше или равен двум тактам HCLK.

30.6.7 Отдельные замечания по режимам ввода/вывода

Чтобы обеспечить некоторую гибкость при конфигурировании режимов ввода/вывода, модуль предоставляет добавочные контакты. Более подробно конфигурирование ядра для работы в XIP режиме описано в руководстве по конфигурации XIP.

1. XIP_CFG_HP_MODE [7:0]: в Dual или Quad режимах ввода/вывода, после того, как адрес передан, устройства flash-памяти QSPI ожидают командного байта, который содержит указание, должно ли устройство дальше работать в стандартном или высокопроизводительном режиме. Этот командный байт разнится для разных устройств, таким образом, эти контакты используются чтобы передать командный байт, соответствующий устройству Flash-памяти, используемому в данный момент. Когда ядро QSPI выходит из XIP-режима, оно так же автоматически выходит из режима высокой производительности, так, чтобы в следующее обращение к устройству оно уже находилось в стандартном режиме работы.

2. XIP_CFG_DUMMY_CYCLES [3:0]: в Dual или Quad режимах ввода/вывода быстрого чтения, на высоких частотах, устройствам flash-памяти QSPI требуется несколько пустых тактов для обращения к данным и возвращения их ядру QSPI. Эти контакты ввода используются для конфигурирования ядра QSPI к тому же количеству пустых тактов, какое использует и устройство flash-памяти QSPI.

Замечание: в определенных режимах, ядро QSPI может быть сконфигурировано только на определенное число пустых тактов.

В режимах быстрого чтения, быстрого чтения с Dual выводом и быстрого чтения с Quad выводом, ядро QSPI может быть сконфигурировано на 8, 16 тактов сигнала.

В Dual режиме ввода/вывода быстрого чтения, ядро QSPI может быть сконфигурировано на 4, 8, 12 или 16 тактов сигнала.

В Quad режиме ввода/вывода быстрого чтения, ядро QSPI может быть сконфигурировано на 2, 4, 6, 8, 12, 14 или 16 тактов сигнала.

3. XIP_CFG_END_DUMMY[1:0]: чтобы определить количество необходимых пустых тактов для выхода из высокопроизводительного состояния для передачи (0xEB) в Quad режиме ввода/вывода быстрого чтения, применяется следующее уравнение:

пустые такты сигнала SPI = 2 * (4 + XIP_CFG_ADDR4 + XIP_CFG_HP_END_DUMMY)

30.6.8 Полнодуплексный режим

На рисунке ниже показаны стандартные подключения верхнего уровня, внешние по отношению к модулю IPC-QSPI-AXI для полнодуплексной реализации. Следует отметить, что для сигналов подключения вывода могут быть необходимы инверторы, если у буферов вывода не подключен активный низкий уровень сигнала.

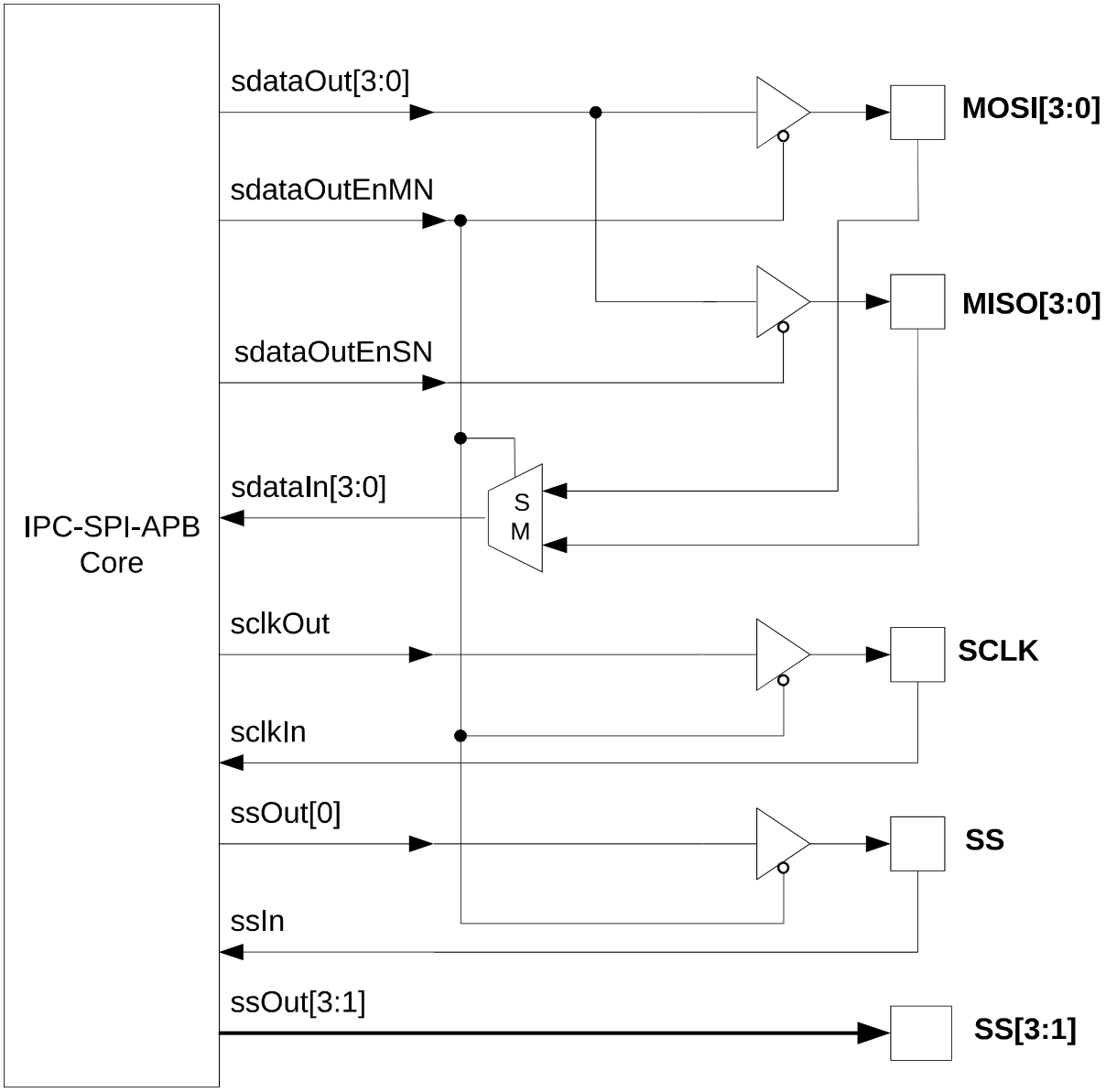


Рисунок 30.8. Типовые подключения ввода/вывода IPC-QSPI-AXI (полнодуплексный режим)

В Quad режиме работы четыре бита являются одновременно вводом и выводом. Старшим битом (MSB) всегда является sdataOut[3]/sdataIn[3]. На диаграмме, sdataOut[3] должен быть подключен к MOSI[3]/MISO[3], и так далее. Аналогично для входных данных, sdataIn[3] должен быть подключен к MISO[3]/MOSI[3].

В Dual режиме работы, sdataOut[1] должен быть подключен к MOSI[1]/MISO[1], а sdataOut[0] должен быть подключен к MOSI[0]/MISO[0].

Аналогично для входных данных, sdataIn[1] должен быть подключен к MISO[1]/MOSI[1], а sdataIn[0] должен быть подключен к MISO[0]/MOSI[0]. sdataOut[3:2] может быть оставлен неподключенным, а sdataIn[3:2] может быть запущен или подтянут к высокому либо низкому уровню.

Для однобитных операций, используются только sdataOut[0] и sdataIn[0]. sdataOut[3:1] может быть оставлен неподключенным, а sdataIn[3:1] может быть запущен или подтянут к высокому либо низкому уровню.

30.6.9 Полудуплексный режим

На рисунке ниже показаны стандартные подключения верхнего уровня, внешние по отношению к модулю IPC-QSPI-AXI для полудуплексной реализации. Следует отметить, что для сигналов подключения вывода могут быть необходимы инверторы, если у буферов вывода не подключен активный низкий уровень сигнала.

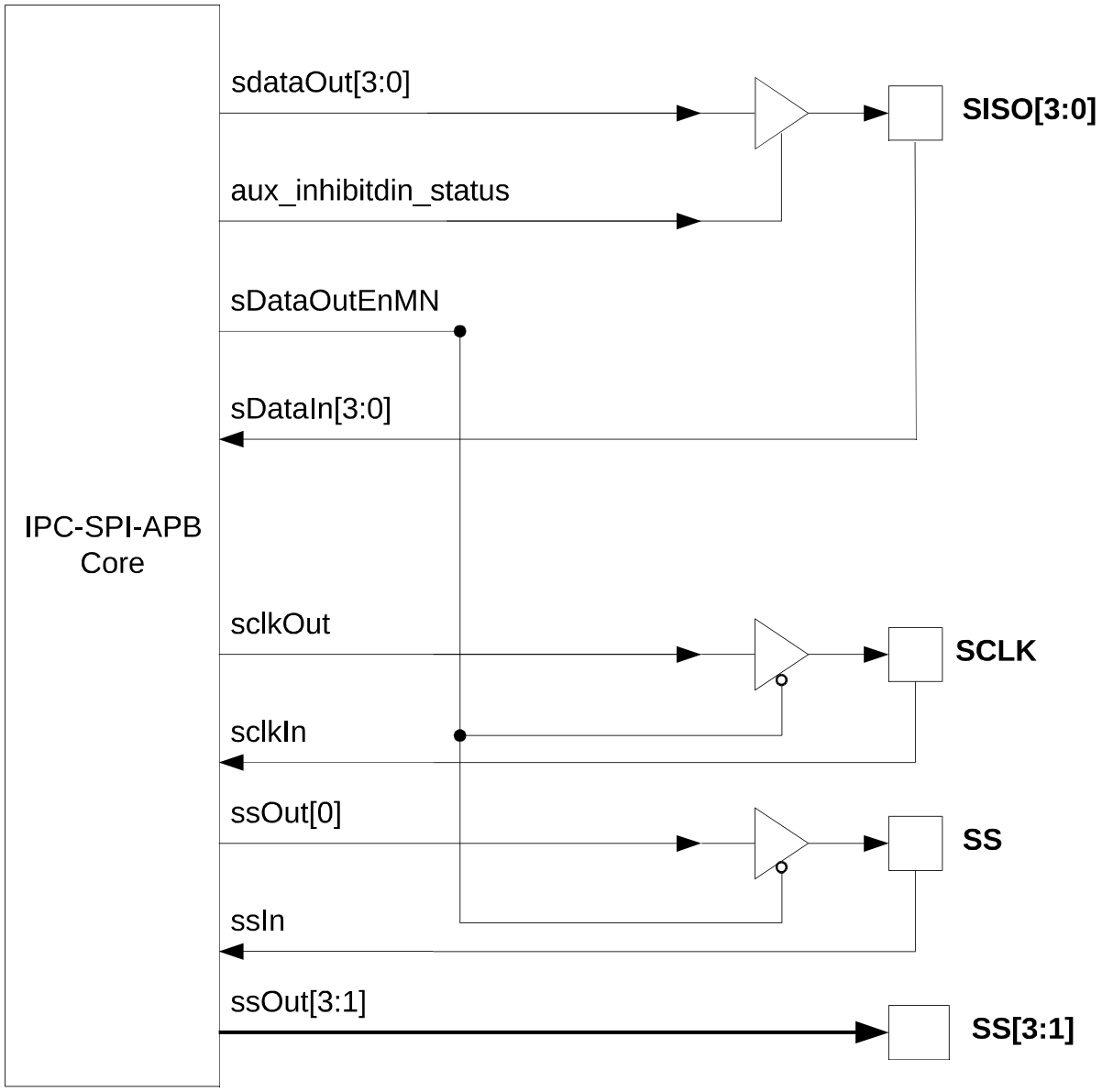


Рисунок 30.9. Типовые подключения ввода/вывода IPC-QSPI-AXI (полудуплексный режим)

Из-за большого разнообразия протоколов, использующих SPI- или QSPI-подобные сигналы чтобы реализовать полудуплексный режим работы, биты вспомогательного регистра управления выводятся от контроллера QSPI для помощи в подобных случаях. Эти биты - aux_inhibitdin_status и aux_inhibitdout_status. Следует отметить что aux_inhibitdin_status указывает что записи в буфер приема RX FIFO от шины QSPI запрещены, в то время как aux_inhibitdout_status указывает что чтения из буфера передачи TX FIFO модулями сериализатора SPI запрещены.

Могут быть также и другие случаи, которые тоже требуют специальных сигналов; для таких случаев, существует 8 выводов общего назначения (spiGpo[7:0]) доступных для специализированной коммутации извне.

В режиме Master-устройства, следует отметить что буфер передачи TX FIFO не должен быть пустым одновременно с тем, как Master-устройство запускает тактовый сигнал, выбор slave-устройства итд. Это выполняется всегда, даже когда шину SISO запускает внешнее slave-устройство. В случае, когда Master-устройство исполняет чтение с внешнего slave-устройства, система Master ответственна за запись пустых данных в буфер передачи TX FIFO на протяжении «чтения» в дополнение к чтению валидных данных, полученных от slave-устройства. Если буфер TX FIFO пуст, Master-устройство не переключит sclkOut и не установит никакие из сигналов ssOut.

В режиме Master-устройства, Master полностью контролирует тайминг передачи, так что совместимость с большим разнообразием slave-устройств, которые требуют разных таймингов, вполне возможна. Однако, в Slave-режиме, slave-устройство должно отвечать master-запросам так, как они установлены; это запросы могут иметь, а могут и не иметь щедрых норм для таймингов на ответ системы, управляющей Slave-устройством. Успешная работа в slave-режиме более вероятна, когда Master-устройство даёт достаточно времени между передачами на то, чтобы Slave-система отреагировала путём загрузки своего буфера TX FIFO валидными данными и контролем направления SISO.

30.7 Описание и подключение вводов/выводов

30.7.1 Список сигналов и описаний

Таблица 30.3. Стандартный интерфейс AXI

Имя сигнала	Тип	Описание
ARESETn	I	Сброс AXI (активный низкий уровень)
ACLK	I	Тактовый сигнал AXI

Таблица 30.4. Интерфейс AXI Slave-устройства без XIP (регистры)

Имя сигнала	Тип	Описание
AWID_R[W:0]	I	Идентификатор адреса записи AXI (W=QSPI_AXI_ID_BITS-1)
AWADDR_R[31:0]	I	Шина адресов записи AXI
AWLEN_R[3:0]	I	Длина пакета записи AXI
AWSIZE_R[2:0]	I	Размер передачи записи AXI
AWBURST_R[1:0]	I	Тип пакета записи AXI
AWLOCK_R[1:0]	I	Блокировка передачи записи AXI
AWCACHE_R[3:0]	I	Данные кэша записи AXI
AWPROT_R[2:0]	I	Данные защиты записи AXI
AWVALID_R	I	Сигнал подтверждения валидности канала адреса записи AXI
AWREADY_R	O	Сигнал подтверждения готовности канала адреса записи AXI
ARID_R[W:0]	I	Идентификатор адреса чтения AXI (W=QSPI_AXI_ID_BITS-1)
ARADDR_R[31:0]	I	Шина адреса чтения AXI
ARLEN_R[3:0]	I	Длина пакета чтения AXI
ARSIZE_R[2:0]	I	Размер передачи чтения AXI
ARWBURST_R[1:0]	I	Тип пакета чтения AXI
ARLOCK_R[1:0]	I	Блокировка передачи чтения AXI
ARCACHE_R[3:0]	I	Данные кэша чтения AXI
ARPROT_R[2:0]	I	Данные защиты чтения AXI
ARVALID_R	I	Сигнал подтверждения валидности канала адреса чтения AXI
ARREADY_R	O	Сигнал подтверждения готовности канала адреса чтения AXI

Имя сигнала	Тип	Описание
WID_R[W:0]	I	Идентификатор данных записи AXI (W=QSPI_AXI_ID_BITS-1)
WDATA_R[W:0]	I	Шина данных записи AXI (W=DSIZE-1)
WSTRB_R[W:0]	I	Строб байтовой линии записи AXI (W=DSIZE/8-1)
WLAST_R	I	Импульс последней передачи данных записи AXI
WVALID_R	I	Сигнал подтверждения валидности канала данных записи AXI
WREADY_R	O	Сигнал подтверждения готовности канала данных записи AXI
BID_R[W:0]	O	Идентификтоа ответа записи AXI (W=QSPI_AXI_ID_BITS-1)
BRESP_R[1:0]	O	Шина ответа записи AXI
BVALID_R	O	Сигнал подтверждения валидности канала ответа записи AXI
BREADY_R	I	Сигнал подтверждения готовности канала ответа записи AXI
RID_R[W:0]	O	Идентификатор данных чтения AXI (W=QSPI_AXI_ID_BITS-1)
RDATA_R[W:0]	O	Данные чтения AXI (W=DSIZE-1)
RRESP_R[1:0]	O	Шина ответа данных чтения AXI
RLAST_R	O	Импульс последней передачи данных чтения AXI
RVALID_R	O	Сигнал подтверждения валидности канала ответа записи AXI
RREADY_R	I	Сигнал подтверждения готовности канала ответа записи AXI

Таблица 30.5. Интерфейс AXI Slave-устройства XIP

Имя сигнала	Тип	Описание
AWID_X[W:0]	I	Идентификатор адреса записи AXI (W=QSPI_AXI_ID_BITS-1)
AWADDR_X[31:0]	I	Шина адреса записи AXI
AWLEN_X[3:0]	I	Длина пакета записи AXI
AWSIZE_X[2:0]	I	Размер передачи записи AXI
AWBURST_X[1:0]	I	Тип пакета записи AXI
AWLOCK_X[1:0]	I	Замыкание передачи записи AXI
AWCACHE_X[3:0]	I	Данные кэша записи AXI
AWPROT_X[2:0]	I	Данные защиты записи AXI
AWVALID_X	I	Сигнал подтверждения валидности канала адреса записи AXI
AWREADY_X	O	Сигнал подтверждения готовности канала адреса записи AXI
ARID_X[W:0]	I	Идентификатор адреса чтения AXI (W=QSPI_AXI_ID_BITS-1)
ARADDR_X[31:0]	I	Шина адреса чтения AXI
ARLEN_X[3:0]	I	Длина пакета чтения AXI
ARSIZE_X[2:0]	I	Размер передачи чтения AXI
ARWBURST_X[1:0]	I	Тип пакета чтения AXI
ARLOCK_X[1:0]	I	Замыкание передачи чтения AXI
ARCACHE_X[3:0]	I	Данные кэша чтения AXI
ARPROT_X[2:0]	I	Данные защиты чтения AXI
ARVALID_X	I	Сигнал подтверждения валидности канала адреса чтения AXI
ARREADY_X	O	Сигнал подтверждения готовности канала адреса чтения AXI
WID_X[W:0]	I	Идентификатор данных записи AXI (W=QSPI_AXI_ID_BITS-1)
WDATA_X[W:0]	I	Шина данных записи AXI (W=DSIZE-1)
WSTRB_X[W:0]	I	Стробы байтовой линии записи AXI (W=DSIZE/8-1)
WLAST_X	I	Импульс последней передачи данных записи AXI
WVALID_X	I	Сигнал подтверждения валидности канала данных записи AXI
WREADY_X	O	Сигнал подтверждения готовности канала данных записи AXI
BID_X[W:0]	O	Идентификатор ответа записи AXI (W=QSPI_AXI_ID_BITS-1)
BRESP_X[1:0]	O	Шина ответа записи AXI
BVALID_X	O	Сигнал подтверждения валидности канала ответа записи AXI
BREADY_X	I	Сигнал подтверждения готовности канала ответа записи AXI
RID_X[W:0]	O	Идентификатор данных чтения AXI (W=QSPI_AXI_ID_BITS-1)
RDATA_X[W:0]	O	Данные чтения AXI (W=DSIZE-1)
RRESP_X[1:0]	O	Шина ответа данных чтения AXI
RLAST_X	O	Импульс последней передачи данных чтения AXI
RVALID_X	O	Сигнал подтверждения валидности канала ответа чтения AXI
RREADY_X	I	Сигнал подтверждения готовности канала ответа чтения AXI

Таблица 30.6. Интерфейс SPI

Имя сигнала	Тип	Описание
extSclkM	I	Тактовый сигнал-источник для тактового сигнала SPI во время Master-режима.
extSclkMN	I	Инвертированный сигнал-источник для тактового сигнала SPI во время Master-режима; этот сигнал должен быть инвертированным extSclkM.
sdataIn[3:0]	I	Последовательный ввод данных SPI. В Quad режиме используются все биты; [3]=msb,[0]=lsb. В Dual режиме используются два бита; [1]=msb, [0]=lsb. В противном случае используется [0].
sdataOut[3:0]	O	Последовательный вывод данных SPI. В Quad режиме используются все биты; [3]=msb,[0]=lsb. В Dual режиме используются два бита; [1]=msb, [0]=lsb. В противном случае используется [0].
sdataOutEnMN	O	Подключение активного низкого вывода Master-устройства sdataOut – применять по протоколу MOSI.
sdataOutEnSN	O	Подключение активного низкого вывода Slave-устройства sdataOut - применять по протоколу MISO.
sclkIn	I	Ввод внешнего последовательного тактового сигнала SPI – Slave-устройство.
sclkInN	I	Инвертированный ввод внешнего последовательного тактового сигнала SPI – Slave-устройство.
ssIn	I	Последовательный ввод переключателя SPI – Slave-устройство.
sclkOut	O	Вывод внешнего последовательного тактового сигнала SPI – Master-устройство.
ssOut[7:0]	O	Вывод выбора Slave-устройства SPI – Master-устройство.

Таблица 30.7. Интерфейс DMA

Имя сигнала	Тип	Описание
dmaBreq[1:0]	O	Запрос пакета DMA. Инициирует пакет. [0] – канал передачи TX (DMA делает записи в TX Fifo) [1] – канал приема RX (DMA делает чтения из RX Fifo)
dmaSreq[1:0]	O	Запрос на единичную передачу DMA. Инициирует единичную передачу. [0] - канал TX, не используется (1’b0) [1] - канал RX (DMA делает чтения из RX Fifo)
dmaClr[1:0]	I	Очистка DMA /Подтверждение от контроллера DMA. [0] - канал TX (DMA делает записи в TX Fifo) [1] - канал RX (DMA делает чтения из RX Fifo)

Таблица 30.8. Интерфейс XIP

Имя сигнала	Тип	Описание
XIP_CFG_CMD[7:0]	I	Тип команды чтения конфигурации XIP. 8'h03=чтение, 8'h0B=быстрое чтение, 8'h3B=быстрое чтение, Dual вывод, 8'h6B=быстрое чтение, Quad вывод, 8'hBB=быстрое чтение, Dual ввод/вывод, 8'hEB=быстрое чтение, Quad ввод/вывод, все другие зарезервированы.
XIP_CFG_HPEN	I	Режим высокой производительности конфигурации XIP. 1'b0=высокопроизводительный режим отключен, 1'b1=высокопроизводительный режим подключен.
XIP_CFG_SSEN[7:0]	I	Подключение выбора Slave-устройства конфигурации XIP 8'b00000001= выбор Slave-устройства 0, 8'b00000010= выбор Slave-устройства 1,

Имя сигнала	Тип	Описание
		..., 8'b10000000=выбор Slave-устройства 7, все остальные зарезервированы.
XIP_CFG_CPHA	I	Режим CPHA конфигурации XIP 1'b0=режим CPHA 0, 1'b1=режим CPHA 1
XIP_CFG_CPOL	I	Режим CPOL конфигурации XIP 1'b0=режим CPOL 0, 1'b1=режим CPOL 1
XIP_CFG_ADDR4	I	4-байтовый режим адресов конфигурации XIP. 1'b0=XIP выпускает 3-байтовый адрес SPI, 1'b1=XIP выпускает 4-байтовый адрес SPI.
XIP_CFG_LE32	I	32-битная организация данных чтения в конфигурации XIP от младшего к старшему. 1'b0=данные чтения организованы от старшего к младшему, 1'b1=данные чтения – 32-битные, от младшего к старшему.
XIP_CFG_HP_MODE [7:0]	I	Командный байт режима XIP. Этот байт передаётся от ядра QSPI к флеш-модулю QSPI. Это значение, заданное разными флеш модулями QSPI, которое следует вводить в их регистры управления для активации режима оптимизации производительности в режимах Dual и Quad ввода/вывода.
XIP_CFG_DUMMY_ CYCLES [3:0]	I	Используется для конфигурирования количества пустых тактов, используемых ядром Qspi. Для режимов быстрого чтения, быстрого чтения с Dual и с Quad выводом, количество пустых тактов равно данному значению, умноженному на 8. Быстрое чтение с Dual вводом/выводом: количество пустых тактов равно данному значению, умноженному на 4 плюс еще 4. Быстрое чтение с Quad вводом/выводом: количество пустых тактов равно данному значению, умноженному на 2 плюс еще 2.
XIP_CFG_HP_END_ DUMMY[1:0]	I	Это значение используется для конфигурирования числа пустых тактов, необходимых для прекращения высокопроизводительного режима. Оно используется ядром при выходе из режима XIP после исполнения высокопроизводительных передач, чтобы удостовериться, что устройство FLASH-памяти не находится всё ещё в высокопроизводительном режиме. Для Dual режима, это значение умножается на 4, чтобы получить нужное реализованное количество пустых тактов. Для Quad режима это значение умножается на 2, чтобы получить нужное реализованное количество пустых тактов.
XIP_EN_REQ	I	Запрос на подключение XIP. Для подключения режима XIP: установить 1'b1, не делать обращений в область адресов QSPI, не менять сигналы XIP_CFG_X, дождаться, пока XIP_EN_OUT примет значение 1'b1. Для отключения режима XIP: установить 1'b0, не делать обращений в область адресов QSPI, не менять сигналы XIP_CFG_X, дождаться, пока XIP_EN_OUT примет значение 1'b0.
XIP_EN_OUT	O	Статус подключения XIP 1'b0=Режим XIP отключен, 1'b1=Режим XIP подключен.

Таблица 30.9. Прочие сигналы

Имя сигнала	Тип	Описание
aux_xfermode_status[1:0]	O	Статус битов Xfermode из вспомогательного регистра управления.
aux_qmode_status[1:0]	O	Статус битов Qmode из вспомогательного регистра управления.
aux_inhibitdin_status	O	Статус бита Inhibitdin из вспомогательного регистра управления.

Имя сигнала	Тип	Описание
aux_inhibitdout_status	O	Статус бита inibitdout из вспомогательного регистра управления.
intReq	O	Запрос прерывания (1 = прерывание, 0 = нет прерываний)
spiGpo[7:0]	O	Выводы общего назначения (домен HCLK)
spiGpi[1:0]	I	Вводы общего назначения, внутренне синхронизированные с доменом HCLK.
scan_en	I	Подключение режима сканирования. Установить в этот сигнал 1'b0 для стандартной работы. Установить в этот сигнал 1'b1 при выполнении сканирования для теста.
extEnable[2:0]	I	Бит подключения QSPI в регистре подключения установлен на положительный фронт этого сигнала. Этот сигнал синхронизирован с HCLK.

30.7.2 Параметры верхнего уровня

Таблица 30.10. Параметры верхнего уровня

Параметр	Возможность обхода	Диапазон	Сброс	Описание
DSIZE	Есть	32,64	32	Ширина шин данных чтения/записи AXI
QSPI_AXI_ID_BITS	Есть	[4,16]	4	Ширина шин идентификаторов AXI (AW, AR, R, W, B)
QSPI_NUM_INTS	Нет	8		Количество условий для прерывания, используемых внутренне для генерации прерывания QSPI.
QSPI_FIFO_WIDTH	Нет	32		Ширина внутренних буферов данных передачи TX /приема RX FIFO.
QSPI_FIFO_ADDR_BITS	Есть	[3,8]	4	Описывает глубину буферов данных TX/RX FIFO 3=глубина 8, 4=глубина 16, 5=глубина 32, 6=глубина 64, итд.
QSPI_XIP_N_ADDR	Нет	25		Максимальное число линий адресов, поддерживаемых логикой XIP, к внешнему устройству FLASH-памяти QSPI.
QSPI_XIP_EN_DEFAULT	Есть	[0,1]	0	Задает работу модуля QSPI по умолчанию сразу после сброса
QSPI_COMPILE_MASTER	Есть	[0,1]	1	В некоторых случаях может быть желательно иметь

Параметр	Возможность обхода	Диапазон	Сброс	Описание
				реализацию QSPI, которая работает ТОЛЬКО как Slave. Если установлен 0, функционал Master-режима исключается из ядра. Если установлена 1, функционал Master-режима включается в ядро.
QSPI_COMPILE_SLAVE	Есть	[0,1]	1	В некоторых случаях может быть желательно иметь реализацию QSPI, которая работает ТОЛЬКО как Master. Если установлен 0, функционал Slave-режима исключается из ядра.

30.8 Описание регистров

30.8.1 Обзор регистров

Таблица 30.11. Смещение регистров

Регистр	Смещение	Описание
QSPI_TX_DATA	0x00	Регистр последовательной передачи данных
QSPI_RX_DATA	0x04	Регистр последовательного приёма данных
	0x08	Зарезервировано
QSPI_CTRL	0x0C	Основной регистр управления
QSPI_CTRL_AUX	0x10	Вспомогательный регистр управления
QSPI_STAT	0x14	Регистр статуса
QSPI_SS	0x18	Регистр выбора Slave-устройства
QSPI_SS_POLARITY	0x1C	Регистр полярности выбора Slave-устройства
QSPI_INTR_EN	0x20	Регистр подключения прерываний
QSPI_INTR_STAT	0x24	Регистр статуса прерываний
QSPI_INTR_CLR	0x28	Регистр очистки прерываний
QSPI_TX_FIFO_LVL	0x2C	Регистр уровня заполнения буфера передачи FIFO (RO – только чтение)
QSPI_RX_FIFO_LVL	0x30	Регистр уровня заполнения буфера приема FIFO (RO – только чтение)
QSPI_UNUSED	0x34	Резерв
QSPI_MASTER_DELAY	0x38	Регистр задержки между передачами в Master-режиме

Регистр	Смещение	Описание
QSPI_ENABLE	0x3C	Регистр подключения/отключения
QSPI_GPO_SET	0x40	Регистр установки выводов общего назначения
QSPI_GPO_CLR	0x44	Регистр очистки выводов общего назначения
QSPI_FIFO_DEPTH	0x48	Регистр сконфигурированной глубины FIFO (RO – только чтение)
QSPI_FIFO_WMARK	0x4C	Регистр предельного уровня TX/RX
QSPI_TX_DUMMY	0x50	Регистр пустой загрузки TX FIFO

30.8.2 Регистр последовательной передачи данных (в Tx FIFO)

Смещение 0x00.

Таблица 30.12. Регистр последовательной передачи данных (в Tx FIFO)

Биты	Сигнал	Тип доступа	Сброс	Описание
[31:0]	data reg	WO	0	Регистр последовательной передачи данных. Попытки упаковки передаваемых данных не предпринимаются. Таким образом, количество переданных на элемент FIFO битов определяется значением “bitsize” вспомогательного регистра управления. К тому же, значение “msb1st” в регистре управления определяет, какие биты элемента FIFO используются: старшие или младшие.

30.8.3 Регистр последовательного приема данных (из Rx FIFO)

Смещение 0x04.

Таблица 30.13. Регистр последовательного приема данных (из Rx FIFO)

Биты	Сигнал	Тип доступа	Сброс	Описание
[31:0]	data reg	RO	0	Регистр последовательного приема данных. Попытки упаковки принимаемых данных не предпринимаются. Таким образом, количество принятых на элемент FIFO битов определяется значением “bitsize” вспомогательного регистра управления.

30.8.4 Резерв

Смещение 0x08 зарезервировано, чтения/записи в эту область не оказывают никакого влияния на работу ядра.

30.8.5 Регистр управления

Смещение 0x0C.

Замечание: глубина txFIFO и rxFIFO может быть задана от 8 до 256 путем установки параметра QSPI_FIFO_ADDR_BITS.

Его значение может быть любой степенью двойки в диапазоне [8,256]. Ширина txFIFO и rxFIFO должна быть сконфигурирована 32-битной путём установки 32 в параметр QSPI_FIFO_WIDTH.

Таблица 30.14. Регистр управления

Биты	Сигнал	Тип доступа	Сброс	Описание
[15:12]	Зарезервировано	R/W	0	Зарезервировано для дальнейшего использования
[11]	mWaitEn	R/W	0	Задержка между передачами для работы в режиме Master. 0=отключение задержки (полная скорость), 1=подключение задержки. См. mWait.
[10]	dma	R/W	0	Установить 1 для подключения режима DMA.
[9:6]	Зарезервировано	R/W	0000	Зарезервировано
[5]	master	R/W	1	Master-режим работы шины порта SPI; 0=slave-режим; 1=master-режим.
[4]	Cpol	R/W	0	0 = тактовый сигнал SPI остается низким, 1 = тактовый сигнал SPI остается высоким.
[3]	cpha	R/W	0	0 = первые биты sdataOut запускаются при установке SS (выбора slave-устройства). 1 = sdataOut запускается на первом фронте тактового сигнала SPI после установки SS.
[2]	msb1st	R/W	0	MSB/LSB 1=старший бит (MSB) первый, 0=младший бит (LSB) первый.

Биты	Сигнал	Тип доступа	Сброс	Описание
				Замечание: в Quad или Dual режиме, эта установка влияет только на порядок (4-битный или 2-битный) кусочной передачи, а не на порядок битов внутри куска; sdataOut[3]/sdataIn[3] всегда являются старшим битом (MSB) куска данных в Quad режиме. sdataOut[1]/sdataIn[1] всегда являются старшим битом (MSB) куска данных в Dual режиме.
[1]	Зарезервировано	R/W	0	Зарезервировано
[0]	contXfer	R/W	1	Бит непрерывной передачи. Это значимо только в Master-режиме. 0=между последовательными передачами бит ssOut становится неактивным. 1=бит ssOut остается активен до тех пор, пока TX FIFO не опустеет И пока бит contXferExtend (во вспомогательном регистре управления) не будет установлен на низкий уровень.

30.8.6 Вспомогательный регистр управления

Смещение 0x10.

Замечание: для Master-режима, вспомогательный регистр управления может быть безопасно переконфигурирован, когда контроллер QSPI подключен, в следующих состояниях: TX FIFO пуст И никаких передач не происходит. Предполагается, что биты [3:0] этой шины могут пригодиться в запуске или конфигурировании внешней шины SPI для различных протоколов SPI/QSPI. По этой причине, биты этого регистра сделаны доступными в виде портов вывода: aux_inhibitdin_status, aux_inhibitdout_status, aux_qmode_status, и aux_xfermode_status.

Таблица 30.15. Вспомогательный регистр управления

Биты	Сигнал	Тип доступа	Сброс	Описание
[12:8]	bitsize	R/W	‘b11111	Bitsize – от 4 до 32 бит ‘h03 = 4 бита ‘h04 = 5 бит ‘h05 = 6 бит ... ‘h1F = 32 бита
[7]	contXferExtend	R/W	0	Бит продления непрервыной передачи.

Биты	Сигнал	Тип доступа	Сброс	Описание
				Используется только в Master-режиме. Более того, этот бит не имеет эффекта если бит contXfer в регистре управления равен 0. Этот бит синхронизирован с доменом SCLK для применения в нём. Следует отметить, что его установка также затрагивает тайминг сигнала подключения вывода master-устройства, sdataOutEnMN. При установке 1 ssOut[x] остается активен даже когда TX FIFO пуст. При установке 0 ssOut[x] становится неактивен когда TX FIFO пуст.
[6]			0	Зарезервировано
[5:4]	xferFormat	R/W	00	Выбор формата SPI. 00=Motorola SPI. 01=зарезервировано. 10=Синхронный последовательный кадр Texas Instruments – только одна линия. 11=Кадр National Microwire – только одна линия.
[3]	inhibitDin	R/W	0	Установка 1 запрещает шине SPI запись в FIFO данных чтения.
[2]	inhibitDout	R/W	0	Установка 1 запрещает сериализаторам SPI чтения из буфера передачи FIFO. Замечание: эта конфигурация предполагается к использованию в Slave-режиме. Во многих случаях Slave-устройство просто получает данные от в Master-устройства; в этих случаях, установка этого бита позволит Slave-устройству принимать данные, не триггеря условие недозаполнения TX FIFO.
[1:0]	qmode	R/W	0	Режимы Spi: 11=Quad SPI, 10=Dual SPI, 01=зарезервировано, 00=стандартный SPI.

30.8.7 Регистр статуса

Смещение 0x14

Таблица 30.16. Регистр статуса

Биты	Сигнал	Тип доступа	Сброс	Описание
[9]	rxTimeout	RO	0	Флаг таймаута приема
[8]	rxOverflow	RO	0	Флаг переполнения FIFO приема
[7]	rxFull	RO	0	Флаг полного FIFO приема
[6]	rxWmark	RO	0	Флаг предельного уровня заполнения FIFO приема
[5]	rxEmpty	RO	1	Флаг пустого FIFO приема
[4]	txFull	RO	0	Флаг полного FIFO передачи
[3]	txWmark	RO	1	Флаг предельного уровня заполнения FIFO передачи
[2]	txEmpty	RO	1	Флаг пустого FIFO передачи
[1]	Зарезервировано	RO	0	Не используется; читать как 0.
[0]	xferIP	RO	0	Передача в процессе исполнения (ssOut/ssIn активны)

30.8.8 Регистр переключения slave-устройства

Смещение 0x18.

Замечание: этот регистр используется только когда контроллер SPI сконфигурирован как Master-устройство SPI.

Таблица 30.17. Регистр переключения slave-устройства.

Биты	Сигнал	Тип доступа	Сброс	Описание
[7]	ssout[7]	R/W	0	0=Не выбирать slave-устройство 7 для TX/RX. 1=Выбрать Slave-устройства 7 для TX/RX.
[6]	ssout[6]	R/W	0	0=Не выбирать slave-устройство 6 для TX/RX. 1=Выбрать Slave-устройства 6 для TX/RX.
[5]	ssout[5]	R/W	0	0=Не выбирать slave-устройство 5 для TX/RX. 1=Выбрать Slave-устройства 5 для TX/RX.
[4]	ssout[4]	R/W	0	0=Не выбирать slave-устройство 4 для TX/RX. 1=Выбрать Slave-устройства 4 для TX/RX.
[3]	ssout[3]	R/W	0	0=Не выбирать slave-устройство 3 для TX/RX. 1=Выбрать Slave-устройства 3 для TX/RX.
[2]	ssout[2]	R/W	0	0=Не выбирать slave-устройство 2 для TX/RX. 1=Выбрать Slave-устройства 2 для TX/RX.
[1]	ssout[1]	R/W	0	0=Не выбирать slave-устройство 1 для TX/RX. 1=Выбрать Slave-устройства 1 для TX/RX.
[0]	ssout[0]	R/W	0	0=Не выбирать slave-устройство 0 для TX/RX.

Биты	Сигнал	Тип доступа	Сброс	Описание
				1=Выбрать Slave-устройства 0 для TX/RX.

Замечание: соответствующий сигнал вывода ssOut[7:0] переключится в активное состояние когда модуль будет подключен и его TX FIFO не пуст. В бите contXfer регистра управления и бите contXferExtend вспомогательного регистра управления приведено описание того, как и когда ssOut[7:0] возвращается в неактивное состояние. Наконец, регистр полярности переключения Slave-устройства задаёт значение «активного» состояния – 1 или 0.

30.8.9 Регистр полярности переключения Slave-устройства

Смещение 0x1C.

Таблица 30.18. Регистр полярности переключения Slave-устройства

Биты	Сигнал	Тип доступа	Сброс	Описание
[7]	Sspol7	R/W	0	0 =ssOut[7] (Master) – активный низкий сигнал. 1= ssOut[7] (Master) – активный высокий сигнал.
[6]	Sspol6	R/W	0	0 =ssOut[6] (Master) – активный низкий сигнал. 1= ssOut[6] (Master) – активный высокий сигнал.
[5]	Sspol5	R/W	0	0 =ssOut[5] (Master) – активный низкий сигнал. 1= ssOut[5] (Master) – активный высокий сигнал.
[4]	Sspol4	R/W	0	0 =ssOut[4] (Master) – активный низкий сигнал. 1= ssOut[4] (Master) – активный высокий сигнал.
[3]	Sspol3	R/W	0	0 =ssOut[3] (Master) – активный низкий сигнал. 1= ssOut[3] (Master) – активный высокий сигнал.
[2]	Sspol2	R/W	0	0 =ssOut[2] (Master) – активный низкий сигнал. 1= ssOut[2] (Master) – активный высокий сигнал.
[1]	Sspol1	R/W	0	0 =ssOut[1] (Master) – активный низкий сигнал. 1= ssOut[1] (Master) – активный высокий сигнал.
[0]	Sspol0	R/W	0	0 =ssOut[0] (Master) или ssIn (Slave) – активный низкий сигнал. 1= ssOut[0] (Master) или ssIn (Slave) – активный высокий сигнал.

30.8.10 Регистр подключения прерывания

Смещение 0x20.

Таблица 30.19. Регистр подключения прерывания.

Биты	Сигнал	Тип доступа	Сброс	Описание
[7:0]	intrEnable	R/W	0	Запись 1 в этот бит подключает отдельные прерывания (см. схему прерываний). Чтение из этого регистра возвращает статус подключенных прерываний.

30.8.11 Регистр статуса прерывания

Смещение 0x24.

Таблица 30.20. Регистр статуса прерывания.

Биты	Сигнал	Тип доступа	Сброс	Описание
[7:0]	intrStatus	RO	0	1 = источник прерываний активен, 0 = прерываний нет.

30.8.12 Регистр очистки прерывания

Смещение 0x28.

Таблица 30.21. Регистр очистки прерывания.

Биты	Сигнал	Тип доступа	Сброс	Описание
[7:0]	intrClear	WClr	0	Очистка прерывания. Запись 1 очищает бит прерывания. Запись 0 не оказывает эффекта.

30.8.13 Схема битов прерывания

Таблица 30.22. Схема битов прерывания.

Биты	Условие	Прерывание с потенциальным запуском
[7]	rxFifoOverflow	Переполнение RX FIFO
[6]	xferDonePulse	Фронт spiGpi[1]
[5]	rxFullPulse	Фронт spiGpi[0]
[4]	rxWmarkPulse	ssOut(Master)/ssIn(Slave)
[3]	rxFullPulse	RX FIFO полон
[2]	rxWmarkPulse	RX FIFO заполнен до предельного уровня
[1]	txWmarkPulse	TX FIFO заполнен до предельного уровня
[0]	txEmptyPulse	TX FIFO пуст

Замечание: актуальный статус флагов FIFO следует смотреть в регистре статусов.

30.8.14 Регистр уровня буфера передачи TX FIFO

Смещение 0x2C.

Таблица 30.23. Регистр уровня буфера передачи TX FIFO.

Биты	Сигнал	Тип доступа	Сброс	Описание
[W:0]	tx_FIFO_level	RO	0	Чтение текущего уровня заполнения буфера передачи FIFO. (W=QSPI_FIFO_ADDR_BITS).

30.8.15 Регистр уровня буфера приема RX FIFO

Смещение 0x30.

Таблица 30.24. Регистр уровня буфера приема RX FIFO.

Биты	Сигнал	Тип доступа	Сброс	Описание
[W:0]	rx_FIFO_level	RO	0	Чтение текущего уровня заполнения буфера приема FIFO. (W=QSPI_FIFO_ADDR_BITS).

30.8.16 Регистр задержки между передачами Master-устройства

Смещение 0x38.

Таблица 30.25. Регистр задержки между передачами Master-устройства.

Биты	Сигнал	Тип доступа	Сброс	Описание
[7:0]	mWait	R/W	0x00	Если master (в регистре управления) имеет значение 1, этот регистр задает задержку между передачами для сериализатора master-устройства. Задержка равна (mwait+1) циклам SCLK.

30.8.17 Регистр подключения

Смещение 0x3C.

Таблица 30.26. Регистр подключения.

Биты	Сигнал	Тип доступа	Сброс	Описание
[3:1]	extEnaSel	R/W	‘b000	‘b000 = установить бит подключения 0 с помощью CPU. ‘b001 = использовать внешнее подключение 0. ‘b010 = использовать внешнее подключение 1. ‘b100 = использовать внешнее подключение 2.

Биты	Сигнал	Тип доступа	Сброс	Описание
				Сигнал extEnable – для запроса подключения (бит 0 регистра подключения).
[0]	enableReq	R/W	0	Подключение SPI. 0= отключение приема и передачи SPI; сброс буферов FIFO RX и TX при переключении этого бита от 1 к 0. Хотя SPI может быть отключен в любое время, рекомендуется делать это, когда буферы FIFO пусты. После записи 0 в бит подключения, следует перепрочесть бит подключения; только после того, как получено чтение того, что бит подключения равен 0, можно производить любые дальнейшие конфигурирования регистров. 1= подключение приема и передачи SPI.

Замечание: чтобы сигнал extEnable подключал передачу и прием SPI, бит подключения [0] должен быть установлен.

Чтобы использовать внутреннее подключение (сигнал enableReq), в регистр подключения следует установить ‘b0001 .

Чтобы использовать внешнее подключение (сигнал extEnable), в регистр подключения следует установить ‘b0011, ‘b0101, или ‘b1001.

Установка в регистр подключения ‘bx0 отключает приём и передачу SPI.

30.8.18 Регистр состояний/установок GPO

Смещение 0x40.

Таблица 30.27. Регистр состояний/установок GPO.

Биты	Сигнал	Тип доступа	Сброс	Описание
[31:0]	gpoSet	R/W	0	Запись: запись единиц (1) устанавливает отдельные биты spiGpo. Чтение: возвращает текущее состояние spiGpo

30.8.19 Регистр очистки GPO/состояний GPI

Смещение 0x44.

Таблица 30.28. Регистр очистки GPO/состояний GPI.

Биты	Сигнал	Тип доступа	Сброс	Описание
[31:0]	gpoClr	R/W	0	Запись: запись единиц (1) очищает отдельные биты spiGpo. Чтение: возвращает текущее состояние spiGpi[1:0].

30.8.20 Регистр глубины буферов приёма(RX)/передачи(TX) FIFO

Смещение 0x48.

Таблица 30.29. Регистр глубины буферов приёма/передачи FIFO.

Биты	Сигнал	Тип доступа	Сброс	Описание
[8:0]	fifoDepth	RO	0	fifoDepth ** Глубина слова от 8 до 256 (степени двойки).

Замечание ** : это значение зависит от параметра “QSPI_FIFO_ADDR_BITS”.

Замечание *** : ширины буферов TX/RX FIFO устанавливаются через параметр “QSPI_FIFO_WIDTH”. Единственное валидное значение - 32.

30.8.21 Регистр предельного уровня TX/RX FIFO

Смещение 0x4C.

Таблица 30.30. Регистр предельного уровня TX/RX FIFO.

Биты	Сигнал	Тип доступа	Сброс	Описание
[15:8]	txWmarkSet	R/W	FIFO_DEPTH/2	Устанавливает значение предельного уровня FIFO передачи.
[7:0]	rxWmarkSet	R/W	FIFO_DEPTH/2	Устанавливает значение предельного уровня FIFO приема.

30.8.22 Регистр пустых записей TX

Смещение 0x50.

Таблица 30.31. Регистр пустых записей TX.

Биты	Сигнал	Тип доступа	Сброс	Описание
[7:0]	txDummyWr	WO	0	Запись значения сюда загружает в буфер TX FIFO N пустых слов данных с содержанием “12341234”. N=значение, например, запись 0x5 загрузит в TX FIFO 5 слов данных со значением ‘1234’. ****

Замечание *** : этот регистр предназначен только для записи. Уровень заполнения TX FIFO может быть прочитан из регистра уровня буфера передачи TX FIFO.

30.9 Тайминги

30.9.1 Тайминг интерфейса AXI

Тайминг интерфейса AXI QSPI зависит от Master-устройства AXI которое инициирует транзакцию. QSPI имеет два AXI-интерфейса, один для стандартной работы/работы с регистрами, и один для работы в режиме XIP. Эти интерфейсы действуют независимо, но только один должен использоваться в отдельный момент времени. Используемый интерфейс должен быть выбран первым.

Обсуждение протокола AXI выходит за рамки данного документа. Подробнее он описан в последней версии документации AXI4 ARM.

30.9.2 Передача данных Master-устройством SPI (обычный режим)

В большинстве случаев различные устройства SPI могут иметь очень разные оценки для таймингов. По этой причине AXI QSPI поддерживает стандартные условия для таймингов, ими управляют два бита регистра управления: cpha и cpol. Эти два бита определяют какие фронты тактового сигнала SPI на каких данных будут переданы и взяты за образец. На диаграммах ниже показана работа этих битов при 8-битной передаче данных.

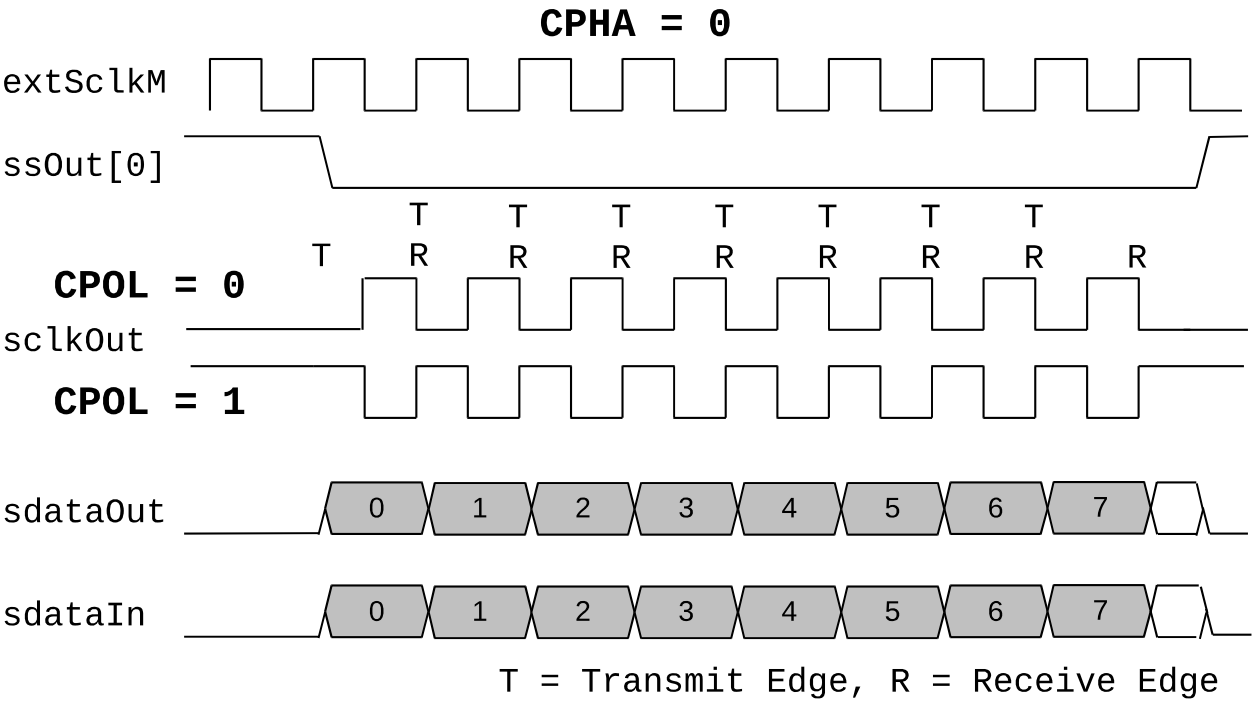


Рисунок 30.10. Передача данных Master-устройства SPI при CPHA = 0, CPOL = 0, 1.

На рисунке выше входной тактовый сигнал extSclkM внешнего Master-устройства SPI используется для производства sclkOut. В регистр управления была произведена запись для выбора Master-режима, и вместе с этим сконфигурированы биты cpol и cpha. Регистр полярности выбора Slave-устройства был очищен для обеспечения сигнала выбора Slave-устройства с активным низким уровнем, а в регистр выбора Slave-устройства была

установлена 1 для подключения взаимодействия со Slave-устройством 0 на шине SPI. В регистр передачи было записано слово данных. Наконец, для обеспечения работы SPI был установлен бит подключения. Данные передаются на нисходящем фронте sclkOut (CPOL = 0), или на восходящем фронте sclkOut (CPOL = 1). Следует отметить, что для CPHA=0, первый бит данных передается как только (или до того как) ssOut становится активен.

Для работы Master-устройства из-за неизбежной задержки распространения возвращенных данных от slave-устройства, лучшим местом для взятия образца входящих данных является полный такт sclkOut после того, как передача состоялась. sclkOut уже является немного запаздывающей версией внутреннего тактового сигнала сериализатора Master-устройства; sclkOut также будет иметь добавочную задержку распространения сигнала из-за выхода за пределы микросхемы к slave-устройству.

Slave-устройство использует этот задержанный тактовый сигнал для производства данных, что накапливает дальнейшую задержку при возвращении обратно в логику master-устройства, которая берёт образец этого сигнала.

Данные должны прибыть и остаться стабильными обратно в логику master-устройства за менее чем один полный такт sclkOut. Такая схема работает куда лучше, чем привязка задержки с круговым маршрутом к половине такта sclkOut. Следует отметить, что этот вариант реализуется, если бы master-устройство брало образец данных на восходящем фронте sclkOut (CPOL=0), или на нисходящем фронте sclkOut (CPOL=1).

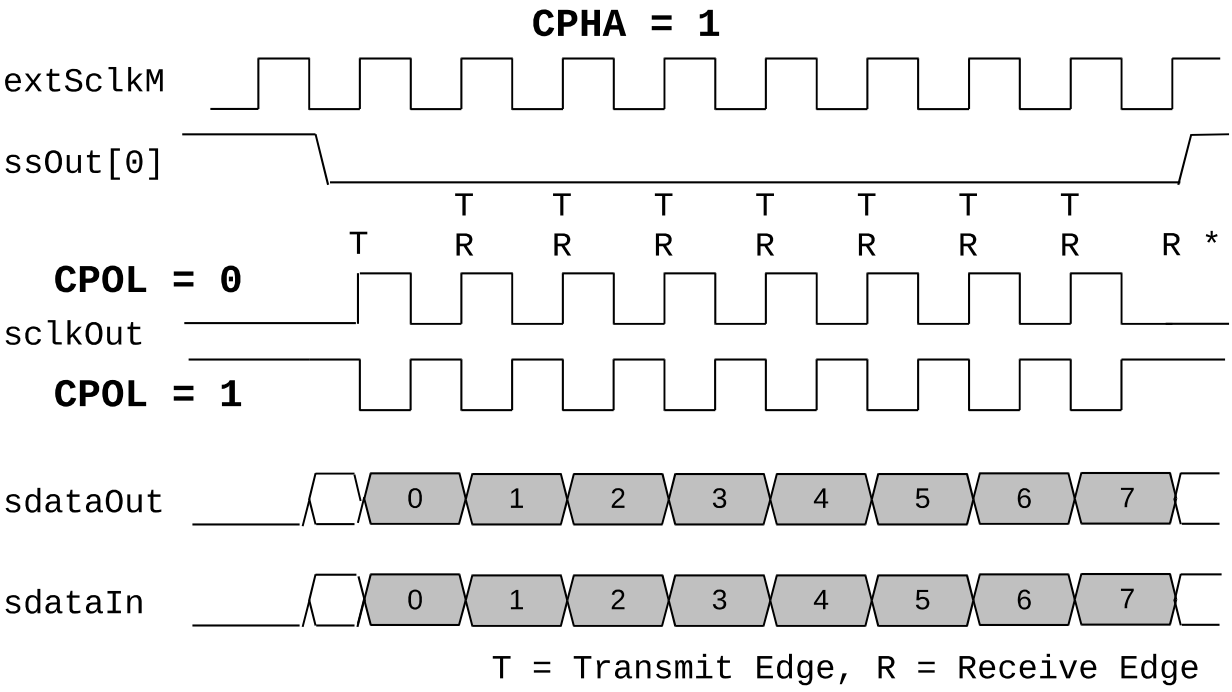


Рисунок 30.11. Передача данных Master-устройства SPI при CPHA = 1, CPOL = 0, 1.

Замечание *: полученные данные сэмплируются внутренним тактовым сигналом master-устройства.

На рисунке выше входной тактовый сигнал extSclkM внешнего Master-устройства SPI используется для производства sclkOut. В регистр управления была произведена запись для выбора Master-режима, и вместе с этим сконфигурированы биты cpol и cpha. Регистр полярности выбора Slave-устройства был очищен для обеспечения сигнала выбора Slave-устройства с активным низким уровнем, а в регистр выбора Slave-устройства была установлена 1 для подключения взаимодействия со Slave-устройством 0 на шине SPI. В регистр передачи было записано слово данных. Наконец, для обеспечения работы SPI был установлен бит подключения. Данные передаются на восходящем фронте sclkOut (CPOL = 0), или на нисходящем фронте sclkOut (CPOL = 1). Следует отметить, что для CPHA=1, первый бит данных передается после того как ssOut становится активен.

Для работы Master-устройства из-за неизбежной задержки распространения возвращенных данных от slave-устройства, лучшим местом для взятия образца входящих данных является полный такт sclkOut после того, как передача состоялась. sclkOut уже является немного запаздывающей версией внутреннего тактового сигнала сериализатора Master-устройства; sclkOut также будет иметь добавочную задержку распространения сигнала из-за выхода вне микросхемы и к slave-устройству.

Slave-устройство использует этот задержанный тактовый сигнал для производства данных, что накапливает дальнейшую задержку при возвращении обратно в логику master-устройства, которая берёт образец этого сигнала.

Данные должны прибыть и остаться стабильными обратно в логику master-устройства за менее чем один полный такт sclkOut. Такая схема работает куда лучше, чем привязка задержки с круговым маршрутом к половине такта sclkOut. Следует отметить, что этот вариант реализуется, если бы master-устройство брало образец данных на восходящем фронте sclkOut (CPOL=0), или на нисходящем фронте sclkOut (CPOL=1).

30.9.3 Передача данных Slave-устройством SPI (обычный режим)

Большая часть рассуждений, касающихся работы в Master-режиме, так же может быть применена и к Slave-режиму. Однако, есть два глобальных отличия.

Первое отличие касается направления отдельных сигналов. В Slave-режиме, тактовый сигнал SPI и сигнал выбора Slave-устройства являются вводами, запускаемыми внешним master-устройством. Логика slave-сериализатора использует sclkIn как свой тактовый сигнал, и использует HCLK-синхронизированный импульс который становится высоким на такте последнего приёма. Когда счётчик принятых данных достигает своего предельного значения (на основе bitsize), полученные данные записываются в буфер приема FIFO и следующее слово данных считывается из буфера передачи FIFO. Это позволяет логике Slave-устройства поддерживать широкий диапазон скоростей тактовых сигналов и Master-устройств. Чтобы учесть задержку синхронизации к домену HCLK, master-устройство не

должно начинать следующую передачу по меньшей мере 2-3 такта HCLK, чтобы slave-устройство имело возможность правильно функционировать.

Второе отличие между работой Slave- и Master-режимов – это то, как сэмплируется sdataIn. В slave-режиме, тактовый сигнал и данные должны иметь одинаковые задержки от Master-устройства. По этой причине, образец данных приёма берётся на полпути между последовательными передачами (на противоположном фронте такта с фронта передачи). Таким образом, данные должны быть стабильны (в середине «глаза» схемы), когда берётся их образец.

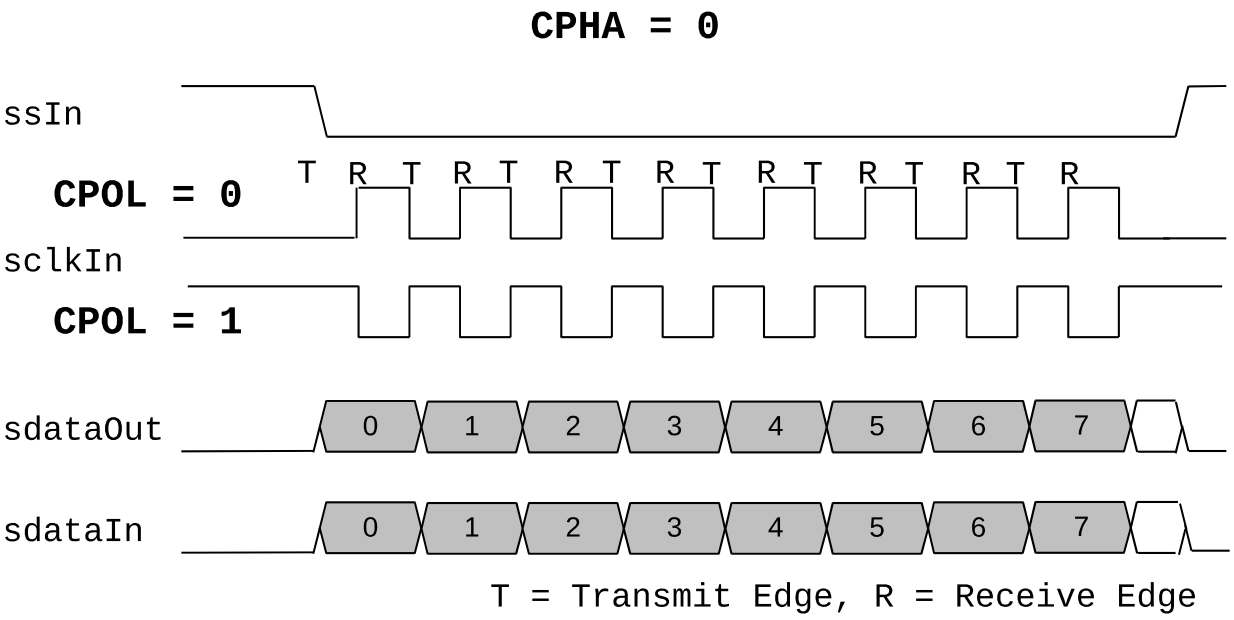


Рисунок 30.12. Передача данных Slave-режима SPI для CPHA = 0, CPOL = 0, 1

Главное что нужно отметить когда CPHA=0, это что sdataOut должен быть валиден до того как появляется тактовый сигнал! Чтобы решить эту проблему, в буфере передачи FIFO применяется память сквозного потока данных (с незарегистрированным выводом). Во время задержки между передачами, следующее слово данных из FIFO уже доступно. Комбинационная логика задаёт подходящие биты, которые могли бы служить выводом на sdataOut. Это приводит к некоторым весьма длительным комбинационным путям из памяти FIFO. Однако, это является проблемой только если время неактивности между передачами очень коротко; логика slave-устройства пользуется периодом неактивности, за который данные распространяются от памяти FIFO к sdataOut.

Данные сдвигаются в регистр сдвига приёма на первом фронте sclkIn следующей передачи. На втором фронте sclkIn, регистр сдвига передачи загружается данными из буфера передачи FIFO, начиная с данных = 1. Финальный фронт тактового сигнала активного периода используется для сброса счетчиков и подготовки логики для вывода верных битов данных под период данных = 0 следующей передачи.

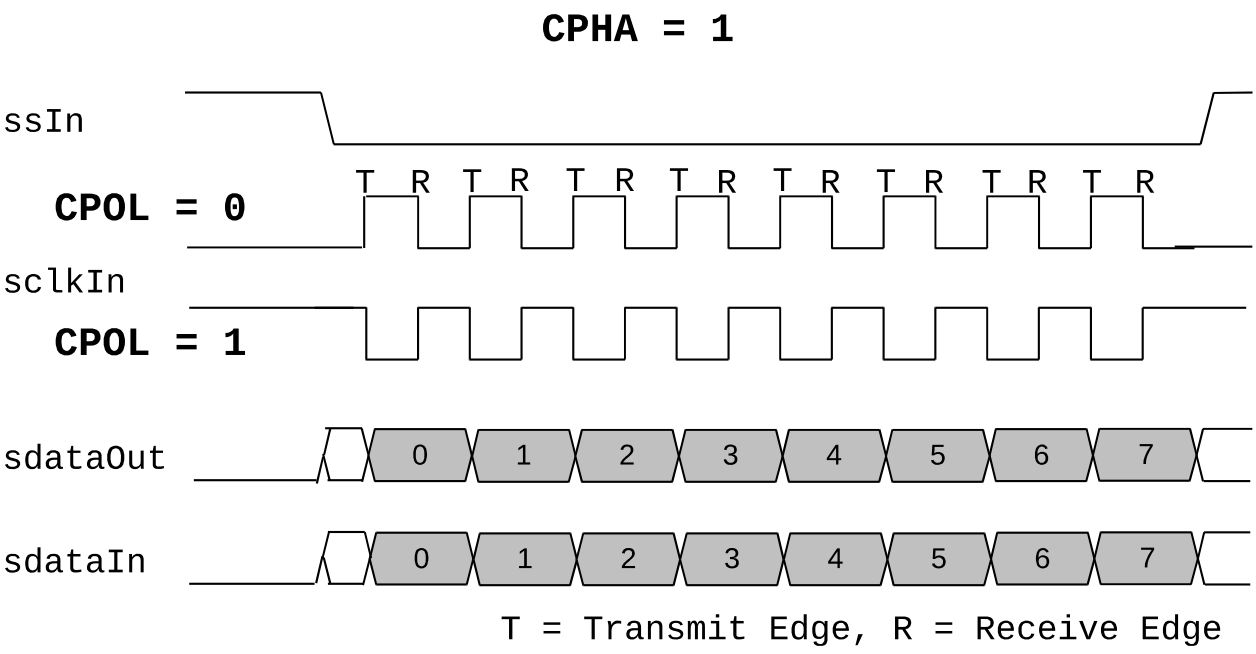


Рисунок 30.13. Передача данных Slave-режима SPI для CPHA = 1, CPOL = 0, 1

Когда CPHA=1, работа slave-устройства более прямолинейна. Когда счетчик данных приёма достигает предельного значения, синхронизированный импульс на домене HCLK продвигает буферы TX и RX. Модуль slave-устройства использует первый фронт тактового сигнала следующей передачи для загрузки регистра сдвига передачи данными из буфера передачи FIFO. После этого момента, каждый другой фронт сигнала используется для взятия образца данных в регистр сдвига приёма, и сдвига данных из регистра сдвига передачи, соответственно.

Наконец, нужно добавить несколько замечаний по работе slave-устройства. В общем случае, импульс, генерируемый для продвижения буферов TX/RX FIFO уходит на высокий потенциал на последнем «чтении» в последовательности, и очищается на втором "чтении" следующей транзакции. Этот импульс синхронизирован с доменом HCLK; восходящий фронт синхронизированного импульса используется для продвижения буферов FIFO. Импульс должен быть крайне широким для того чтобы быть обнаруженным доменом HCLK в поддерживаемых HCLK/SCLK диапазонах частот. Однако, существуют угловые случаи, где ширина этого импульса по необходимости сужается, что еще больше ограничивает соответствующие частоты HCLK/SCLK.

Ниже описываются эти угловые случаи.

В режиме QSPI, когда numbits = 3'b001 (8 бит), импульс, генерируемый для продвижения TX/RX FIFO уходит на высокий уровень на последнем «чтении» в последовательности, и очищается на первом «чтении» следующей транзакции. Это происходит потому, что на транзакцию отведено всего два периода sclk. В этом случае, fHCLK должна быть равна как минимум 1 * fSCLK.

Режим QSPI, когда numbits = 3'b000 (4 бита) является самым экстремальным случаем. Только один период отведён sclk на транзакцию. Поэтому необходимо использовать синхронизированную версию sclk на домене HCLK, чтобы продвинуть буферы TX/RX FIFO в этом случае. Тогда fHCLK должна быть равна как минимум 2 * fSCLK.

30.9.4 Формат синхронного последовательного кадра Texas Instruments

Для формата синхронного последовательного кадра Texas Instruments, контакт ssOut имеет верхний уровень в течение одного такта в начале кадра (восходящий фронт) перед передачей данных. Для этого формата кадров, как master, так и slave-устройство активируют свои данные вывода передачи на восходящем фронте sclkOut, и защёлки получают входные данные на отрицательном фронте sclkOut.

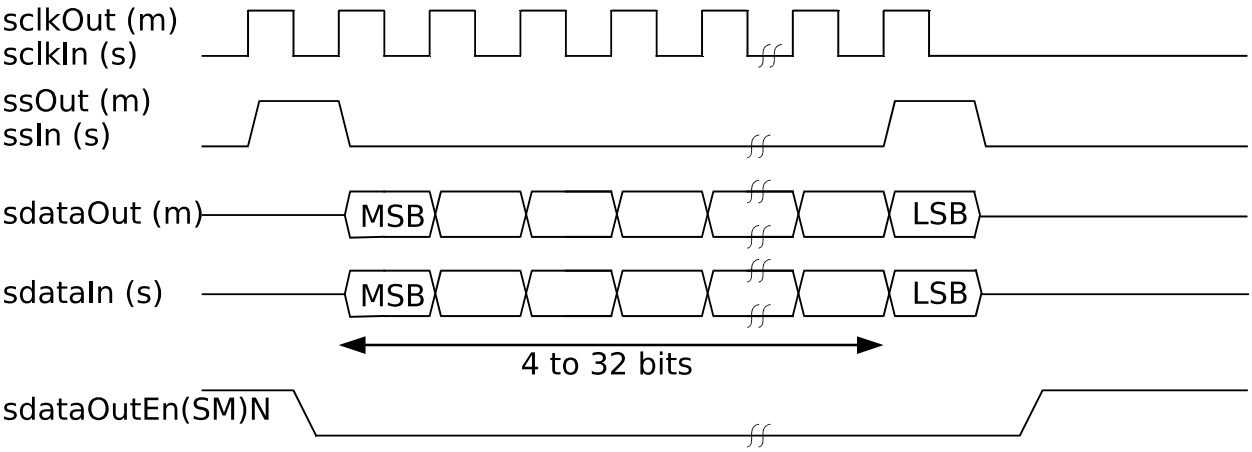


Рисунок 30.14. Единичная передача формата синхронного последовательного кадра TI

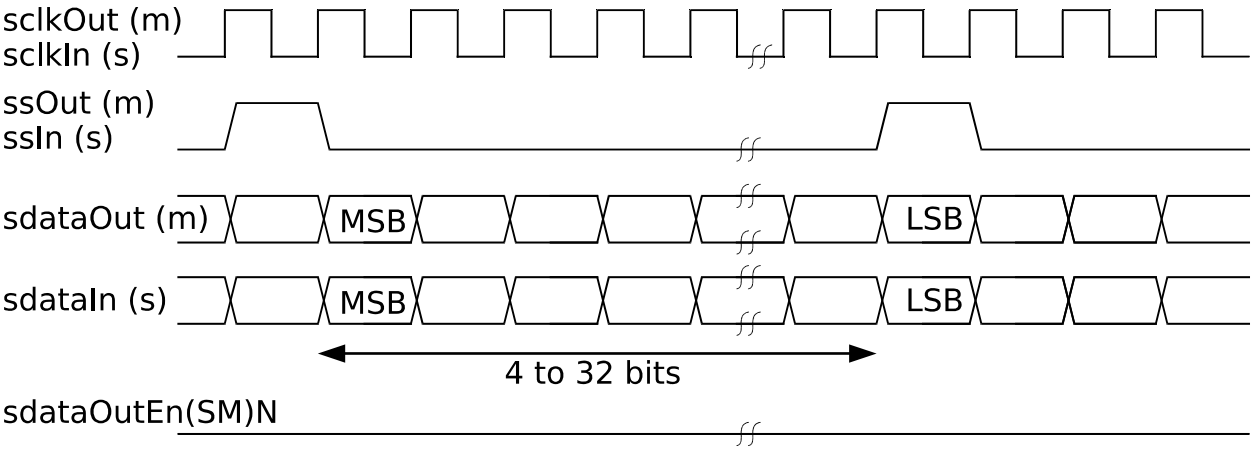


Рисунок 30.15. Последовательная передача формата синхронного последовательного кадра TI

30.9.5 Формат кадра National Microwire

Формат National Microwire аналогичен формату SPI, за исключением того, что передача полудуплексная вместо полнодуплексной. Master-устройство передаёт 8-битное командное слово, ожидает в течение одного цикла тактового сигнала и затем получает данные по сигналу sdataIn. После этого slave-устройство передаст от 4 до 16 битов обратно к master-устройству. Тайминг указан ниже. Master-устройство передаёт sdataOut на нисходящем фронте sclkOut и сэмплирует полученные данные на восходящем фронте sclkOut.

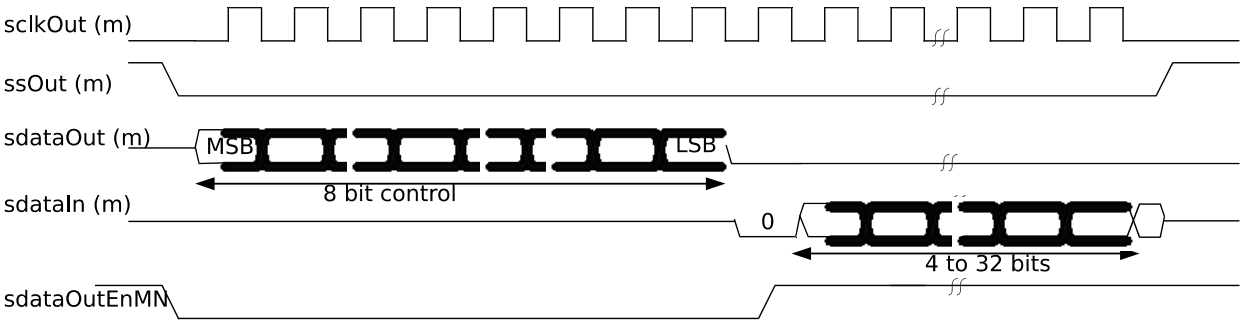


Рисунок 30.16. Единичная передача формата National Microwire

30.9.6 Сигналы DMA

Функционал DMA валиден и для Master и для Slave-режима. Предполагается, что в системе AXI/AXI присутствует внешний контроллер DMA, который подключен к модулю SPI. Таймингом для начала отправки сигналов является простая процедура подтверждения установления связи. Контроллер IPC-QSPI-AXI устанавливает dmaBreq[0], когда его буфер передачи FIFO опустошается до предельного уровня, и устанавливает dmaBreq[1] когда его буфер приема FIFO заполняется до своего предельного уровня. Эти сигналы остаются установленными до тех пор, пока контроллер DMA не совершит достаточного количеств чтений или записей. Достаточное количество чтений/записей - это число, указанное как предельный уровень буферов приема/передачи. Когда контроллер DMA завершает записи в буфер передачи TX FIFO, он устанавливает dmaClr[0]; когда чтения из RX FIFO завершены, контроллер DMA устанавливает dmaClr[1]. Когда IPC-QSPI-AXI обнаруживает dmaClr, он деактивирует соответствующий сигнал dmaBreq. Процедура подтверждения связи завершается когда контроллер DMA сэмплирует низкий уровень dmaBreq и деактивирует dmaClr.

Ниже приведены некоторые дополнительные замечания о сигналах DMA.

IPC-QSPI-AXI устанавливает dmaBreq[x] только если dmaClr[x] имеет низкий уровень, и деактивирует dmaBreq[x], только если dmaClr[x] имеет высокий уровень. Это правило также применимо к dmaSreq. Следует отметить что dmaSreq[1] устанавливается только для состояния таймаута DMA, и что dmaSreq[0] при этом не используется. Наконец, dmaSreq[1] и dmaBreq[1] не устанавливаются одновременно.

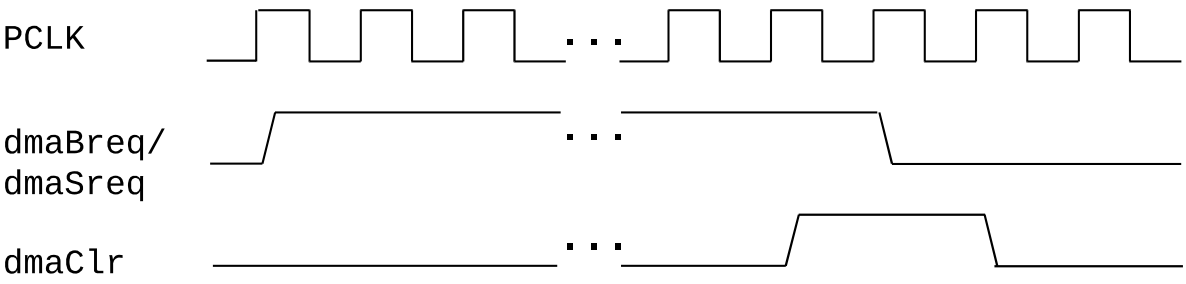


Рисунок 30.17. Сигналы DMA

30.10 Программирование

30.10.1 Подключение и отключение

Установка бита подключения (bit[0]) в регистре подключения включает выполнение транзакций на шине SPI. Очистка бита подключения в регистре подключения сбрасывает порт SPI путём установки, что внешняя линия выбора slave-устройства возвращается в неактивное состояние. Переключение бита подключения из состояния 1 в 0 также провоцирует синхронную очистку буферов FIFO передачи и приёма, и заставляет все поля стаутса FIFO в интерфейсе регистра указывать что буфера FIFO пусты. Отключение ядра не модифицирует никакие другие регистры конфигурации.

30.10.2 Режим XIP

Вход и выход в режим XIP должен осуществляться следующим образом:

1) В идеале, никакие активные команды на чтение/запись не должны выходить в пространство адресов QSPI в процессе входа или выхода из режима XIP.

2) Если режим XIP неактивен, XIP_EN_REQ и XIP_EN_OUT оба имеют значение 0.

3) В параметрах XIP_CFG_X производятся любые изменения конфигурации.

4) Для входа в режим XIP следует запустить XIP_EN_REQ = 1, затем дождаться, когда XIP_EN_OUT = 1.

5) После того, как установится XIP_EN_OUT = 1, обращения на чтение от пространства адресов QSPI будут реализованы в формате одной из нескольких поддерживаемых общепринятых транзакций чтения к устройству FLASH-памяти (Q)SPI. Записи в пространство адресов QSPI в режиме XIP будут возвращать HRESP = ERROR.

6) Для выхода из режима XIP, следует задать XIP_EN_REQ = 0, затем дождаться, когда установится XIP_EN_OUT = 0, прежде чем предпринимать любые дальнейшие обращения к пространству адресов QSPI.

30.10.3 Последовательность запуска

1. Произвести запись в регистр подключения для отключения порта SPI.
2. Произвести чтение регистра подключения – он должен возвращать 0, прежде чем можно будет выполнить следующий шаг.
3. Обнулить любые прерывания путём записи ‘h1FF в регистр очистки прерываний.
4. Если прерывания используются, подключить прерывания путём записи в регистр подключения прерываний.
5. Задать полярность шин переключения путём записи в регистр переключения полярности slave-устройства.
6. Если используется как master-устройство, выбрать какое slave-устройство должно быть выбрано путём записи в регистр выбора slave-устройства.
7. Если используется как master-устройство, и если требуется большая задержка между передачами, для работы с низкопроизводительными slave-устройствами, сделать запись в регистр задержек между передачами Master-устройства.
8. Если используется как master-устройство, и если требуется работа с DMA, сделать запись в регистр таймаута DMA для установки длительности таймаута DMA.
9. Запись в регистр управления:
   а) Подключение задержки между передачами (для master-режима).
   б) Подключение DMA (dmaEnable).
   в) Выбор режима master или slave-устройства.
   г) Установка бита cpol.
   д) Установка бита cpha.
   е) Установка бита msb1st.
   ж) Подключение непрерывной передачи (для master-режима).
10. Запись во вспомогательный регистр управления:
    а) Расширение непрерывной передачи (для master-режима).
    б) Установка бита bitsize (4, 8, 12, 16, 20, 24, 28, 32 бита).
    в) Запрет записей в RX FIFO.
    г) Режим Spi (Quad, Dual, Spi).

11. Запись в регистр подключения для подключения порта SPI.

Следует отметить, что допускается производить предварительные записи в TX FIFO перед подключением модуля в последнем шаге.

Следует также отметить что дальнейшие записи во вспомогательный регистр управления после подключения этого модуля допускаются, если одновременно выполняется следующее: буфер TX FIFO пуст И процесс передачи не происходит.

30.10.4 Последовательность передачи

Загрузка регистра передачи может быть осуществлена разными способами: использование процессора для опроса регистра статуса о состоянии буфера FIFO, использование возможностей прерываний процессора и кода ISR для прерывания когда буфер передачи TX FIFO пуст или заполнен ниже предельного уровня, или использование контроллера DMA (и связанных с ним сигналов) для выполнения необходимых записей в буфер передачи TX FIFO.

Для функционала опроса:

Следует прочесть регистр уровня FIFO, чтобы определить число байтов/слов, которые возможно записать.

Записать байты/слова в регистр последовательной передачи данных.

Прочесть регистр статуса чтобы проверить флаг заполненности буфера передачи TX FIFO перед записью добавочных байтов/слов.

Для функционала прерывания:

Использовать служебную программу прерывания для того чтобы отправить прерывание когда TX FIFO пуст или заполнен ниже предельного уровня.

Записать подходящее число слов данных в регистр последовательной передачи данных в служебной программе прерывания.

Для функционала DMA:

Подключить функционал DMA и позволить подключенному механизму DMA передавать данные в TX FIFO когда dmaBreq[0] установлен в логике DMA.

Для функционала Master-устройства выбор другого slave-устройства осуществляется следующим образом:

Прочесть регистр статуса чтобы удостовериться что TX FIFO пуст и не находится в процессе передачи.

Сделать запись в регистр выбора Slave-устройства (и регистр выбора полярности Slave-устройства, если необходимо) для выбора другого slave-устройства под следующие транзакции.

30.10.5 Последовательность приема

При сигнале о полном FIFO RX/получении прерывания о достижении предельного уровня заполнения FIFO RX или при чтении регистра статуса:

- если установлен флаг rx_not_empty, тогда следует чтение слов(а) данных из регистра последовательного приема данных;
- если применяются прерывания, следует очистить прерывания путем записи «1» в регистр очистки прерываний.

Если применяется режим DMA, нет необходимости читать регистр статуса. Вместо этого, логика DMA установит dmaBreq[1] когда буфер приема RX FIFO заполнится до своего предельного уровня. Если в буфере приема после периода таймаута остались символы, логика DMA установит dmaSreq[1] до тех пор пока контроллер DMA не прочтет эти символы по одному из буфера приема RX FIFO.

30.10.6 Замечания по полудуплексному исполнению последовательностей

Для Master-режима настоятельно рекомендуется менять направление шины данных SISO только когда буфер передачи TX FIFO пуст. Следует отметить что ssOut[x] заведомо переключается в неактивное состояние между успешными передачами (contXfer = 0) или когда уровень TX FIFO пуст (contXfer = 1 и contXferExtend = 0). Для неограниченного расширения периода активности ssOut[x], следует установить contXfer = 1 и contXferExtend = 1. Бит contXferExtend во вспомогательном регистре управления может быть переконфигурирован в любое время под планирование окончания активной передачи. Однако, следует соблюдать осторожность и не модифицировать никакие другие биты во вспомогательном регистре управления во время активной передачи.

Для Master-режима, при «передаче» данных в полудуплексном режиме, желательно запретить записи в буфера приема RX FIFO. И хотя это обычно связано только с полудуплексным режимом, в полнодуплексном режиме тоже могут быть моменты, когда записи в RX FIFO желательно запретить. По этой причине, во вспомогательном регистре управления существует отдельный бит под это (inhibitDin).

Для Master-режима, при «приеме» данных в полудуплексном режиме, необходимо записывать пустые данные в TX FIFO для получения реальных данных от slave-устройства. Пустые данные не передаются на шину SPI (поскольку Master является принимающим

устройством), но используются для отслеживания того, сколько передач «приема» необходимо исполнить. Для каждой записи в TX FIFO, Master-устройство генерирует единичную транзакцию на шину SPI. Система может читать RX FIFO Master-устройства независимо для обращения к полученным данным.

Для Slave-режима настоятельно рекомендуется менять направление шины данных SISO только когда SS (выбор Slave-устройства) неактивен. Некоторые протоколы могут требовать, что направление SISO должно меняться, когда SS активен; в этих случаях, управляющее Master-устройство должно предоставить Slave-системе адекватный промежуток времени между передачами для перемены направления SISO.

Для Slave-режима, при «приеме» данных в полудуплексном режиме, в TX FIFO Slave-устройства не нужно записывать пустые данные, чтобы Slave-устройство могло получить данные. Однако, следует отметить, что отсутствие записей в TX FIFO может в конце концов спровоцировать ошибку обнуления буфера TX FIFO, если внешнее Master-устройство продолжает делать запросы на транзакции.

31. КОНТРОЛЛЕР I2S (I2S)

31.1 Общая информация

Контроллер интерфейса последовательной шины I2S имеет следующие характеристики:

1. 2 стереоканала передачи данных.
2. 1 стереоканал приема данных.
3. Разрешение аудио сигнала до 32 бит.
4. Глубина буферов FIFO приемника/передатчика 16 слов.
5. Аппаратный интерфейс запросов DMA-контроллеру.
6. Программируемый порог заполнения буферов FIFO на прерывание.
7. Выбор источника интерфейсной частоты:
   1) PLLCTR подсистемы lsperiph1_subs.
   2) КП микросхемы.

   Выбор производится через системные регистры lsperiph1_subs.

31.2 Регистры

В Таблица 31.1 приведен перечень программно-доступных регистров контроллера I^2S.

Таблица 31.1. Перечень программно-доступных регистров контроллера I^2S

Условное обозначение Регистра	Название регистра	Тип доступа	Исходное Состояние	Смещение
IER	Регистр включения контроллера.	W/R	0x0	0x0
IRER	Регистр включения блока приема.	W/R	0x0	0x4
ITER	Регистр включения блока передачи.	W/R	0x0	0x8
CER	Регистр включения тактовой частоты.	W/R	0x0	0xC
CCR	Регистр конфигурации тактовой частоты.	W/R	0x10	0x10
RXFFR	Регистр сброса FIFO буферов блока приема.	W	0x0	0x14
TXFFR	Регистр сброса FIFO буферов блока передачи.	W	0x0	0x18
LRBRx	Регистр «левого» слова приемника.	R	0x0	0x20 + 0x40*x

Условное обозначение Регистра	Название регистра	Тип доступа	Исходное Состояние	Смещение
LTHRx	Регистр «левого» слова x - передатчика.	W	0x0	0x20 + 0x40*x
RRBRx	Регистр «правого» слова приемника.	R	0x0	0x24 + 0x40*x
RTHRx	Регистр «правого» слова x - передатчика.	W	0x0	0x24 + 0x40*x
RERx	Регистр включения приемника.	W/R	0x1	0x28 + 0x40*x
TERx	Регистр включения x - передатчика.	W/R	0x1	0x2C + 0x40*x
RCRx	Регистр конфигурации приемника.	W/R	0x5	0x30 + 0x40*x
TCRx	Регистр конфигурации x - передатчика.	W/R	0x5	0x34 + 0x40*x
ISRx	Регистр статуса прерывания x - канала.	R	0x10	0x38 + 0x40*x
IMRx	Регистр маски прерывания x - канала.	W/R	0x33	0x3C + 0x40*x
RORx	Регистр переполнения FIFO приемника.	R	0x0	0x40 + 0x40*x
TORx	Регистр переполнения FIFO x - передатчика.	R	0x0	0x44 + 0x40*x
RFCR0	Регистр конфигурации FIFO приемника.	W/R	0x3	0x48 + 0x40*x
TFCRx	Регистр конфигурации FIFO x - передатчика.	W/R	0x3	0x4C + 0x40*x
RFFx	Регистр сброса FIFO приемника.	W	0x0	0x50 + 0x40*x
TFFx	Регистр сброса FIFO x - передатчика.	W	0x0	0x54 + 0x40*x
RXDMA	Регистр DMA приема.	R	0x0	0x1C0
RRXDMA	Регистр сброса DMA приема.	W	0x0	0x1C4
TXDMA	Регистр DMA передачи.	W	0x0	0x1C8
RTXDMA	Регистр сброса DMA передачи.	W	0x0	0x1CC
I2S_COMP_PARAM_2	Второй регистр параметров компонента.	R	0x48c	0x1F0
I2S_COMP_PARAM_1	Первый регистр параметров компонента.	R	0x264027e	0x1F4
I2S_COMP_VERSION	Регистр ревизии компонента.	R	0x3131302a	0x1F8
I2S_COMP_TYPE	Регистр типа компонента.	R	0x445701a0	0x1FC

31.2.1 Регистр IER

Регистр включения контроллера I^2S.

Формат регистра IER приведен в Таблица 31.2.

Таблица 31.2. Формат регистра IER

Номер бита	Условное Обозначение	Назначение
0	IEN	Установка этого бита в 0 приводит к выключению всех каналов обмена и стиранию содержимого всех FIFO. 1: Контроллер I²S включен. 0: Контроллер I²S выключен.
1:31	-	Резерв

31.2.2 Регистр IRER

Регистр включения блока приема контроллера I²S.

Формат регистра IRER приведен в Таблица 31.3.

Таблица 31.3. Формат регистра IRER

Номер бита	Условное Обозначение	Назначение
0	RXEN	1: Блок приема контроллера I²S включен. 0: Блок приема контроллера I²S выключен.
1:31	-	Резерв

31.2.3 Регистр ITER

Регистр включения блока передачи контроллера I²S.

Формат регистра ITER приведен в Таблица 31.4.

Таблица 31.4. Формат регистра ITER

Номер бита	Условное обозначение	Назначение
0	RXEN	1: Блок передачи контроллера I²S включен. 0: Блок передачи контроллера I²S выключен.
1:31	-	Резерв

31.2.4 Регистр CER

Регистр включения тактовых сигналов SCLK_EN, SCLK_GATE, WS_OUT.

Формат регистра CER приведен в Таблица 31.5.

Таблица 31.5. Формат регистра CER

Номер бита	Условное обозначение	Назначение
0	CLKEN	1: сигналы SCLK_EN, SCLK_GATE, WS_OUT генерируются. 0: тактовые сигналы SCLK_EN, SCLK_GATE, WS_OUT не генерируются (идут низким уровнем).
1:31	-	Резерв

31.2.5 Регистр CCR

Регистр конфигурации генерации тактовых сигналов.

Формат регистра CCR приведен в Таблица 31.6.

Таблица 31.6. Формат регистра CCR

Номер бита	Условное обозначение	Назначение
0:2	SCLKG	Это поле определяет момент перехода сигнала SCLK_GATE из 0 в 1. 0: SCLK_GATE=0 всегда. 1: Через 12 тактов SCLK. 2: Через 16 тактов SCLK. 3: Через 20 тактов SCLK. 4: Через 24 тактов SCLK. Перед изменением этого параметра необходимо выключить генерацию тактовых сигналов (CER[0]=0).
3:4	WSS	Это поле определяет продолжительность высокого и низкого уровней сигнала выбора слова WS_OUT: 0: 16 тактов SCLK. 1: 24 тактов SCLK. 2: 32 тактов SCLK. Перед изменением этого параметра необходимо выключить генерацию тактовых сигналов (CER[0]=0).
5:31	-	Резерв

31.2.6 Регистр RXFFR

Регистр сброса всех FIFO блока приема.

Формат регистра RXFFR приведен в Таблица 31.7.

Таблица 31.7. Формат регистра RXFFR

Номер бита	Условное обозначение	Назначение
0	RXFFR	Запись единицы в этот регистр приводит к стиранию данных во всех FIFO блока приема. Перед записью в этот регистр необходимо выключить блок приема (IRER[0]=0).
1:31	-	Резерв

31.2.7 Регистр TXFFR

Регистр сброса всех FIFO блока передачи.

Формат регистра TXFFR приведен в Таблица 31.8.

Таблица 31.8. Формат регистра TXFFR

Номер бита	Условное обозначение	Назначение
0	TXFFR	Запись единицы в этот регистр приводит к стиранию данных во всех FIFO блока передачи. Перед записью в этот регистр необходимо выключить блок передачи (ITER[0]=0).
1:31	-	Резерв

31.2.8 Регистр LRBRx (for x=0; x <= I2C_RX_CHANNELS-1)

Регистр чтения «левого» слова из приемника.

Формат регистра LRBRx приведен в Таблица 31.9.

Таблица 31.9. Формат регистра LRBRx

Номер бита	Условное обозначение	Назначение
0:31	LRBRx	Через этот регистр происходит чтение «левого» слова данных из FIFO приемника. Прежде чем повторить эту операцию необходимо прочитать «правое» слово данных из FIFO приемника через регистр RRBRx. Размер принимаемого слова описан в регистре RCRx.

31.2.9 Регистр LTHRx (for x=0; x <= I2C_TX_CHANNELS-1)

Регистр «левого» слова передатчика x.

Формат регистра LTHRx приведен в Таблица 31.10.

Таблица 31.10. Формат регистра LTHRx

Номер бита	Условное обозначение	Назначение
0:31	LTHRx	Через этот регистр происходит запись «левого» слова данных в FIFO передатчика x. Прежде чем повторить эту операцию необходимо записать «правое» слово данных в FIFO передатчика x через регистр RTHRx. Размер передаваемого слова описан в соответствующем регистре TCRx.

31.2.10 Регистр RRBRx (for x=0; x <= I2C_RX_CHANNELS-1)

Регистр чтения «правого» слова из приемника.

Формат регистра RRBRx приведен в Таблица 31.11.

Таблица 31.11. Формат регистра RRBRx

Номер бита	Условное обозначение	Назначение
0:31	RRBRx	Через этот регистр происходит чтение «правого» слова данных из FIFO приемника. Этой операции должно предшествовать чтение «левого» слова данных из FIFO приемника через регистр LRBRx. Размер принимаемого слова описан в регистре RCRx.

31.2.11 Регистр RTHRx (for x=0; x <= I2C_TX_CHANNELS-1)

Регистр «правого» слова передатчика x.

Формат регистра RTHRx приведен в Таблица 31.12.

Таблица 31.12. Формат регистра RTHRx

Номер бита	Условное обозначение	Назначение
0:31	RTHRx	Через этот регистр происходит запись «правого» слова данных в FIFO передатчика x. Этой операции должна предшествовать запись «левого» слова данных в FIFO передатчика x через регистр LTHRx.

Номер бита	Условное обозначение	Назначение
		Размер передаваемого слова описан в соответствующем регистре TCRx.

31.2.12 Регистр RERx (for x=0; x <= I2C_RX_CHANNELS-1)

Регистр включения приемника.

Формат регистра RERx приведен в Таблица 31.13.

Таблица 31.13. Формат регистра RERx

Номер бита	Условное обозначение	Назначение
0	RXCHENx	Бит включает приемник. При включении, прием данных начинается с «левого» слова. Глобальное выключение контроллера (RER[0]=0) или блока приема (IRER[0]=0) переписывает значение этого бита. 1: Приемник включен. 2: Приемник выключен.
1:31	-	Резерв

31.2.13 Регистр TERx (for x=0; x <= I2C_TX_CHANNELS-1)

Регистр включения передатчика x.

Формат регистра TERx приведен в Таблица 31.14.

Таблица 31.14. Формат регистра TERx

Номер бита	Условное обозначение	Назначение
0	TXCHENx	Бит включает передатчик x. При включении, передачаданных начинается со следующего «левого» слова. Глобальное выключение контроллера (TER[0]=0) или блока передачи (ITER[0]=0) переписывает значение этого бита. 1: Передатчик x включен. 2: Передатчик x выключен.
1:31	-	Резерв

31.2.14 Регистр RCRx (for x=0; x <= I2C_RX_CHANNELS-1)

Регистр конфигурации приемника.

Формат регистра RCRx приведен в Таблица 31.15.

Таблица 31.15. Формат регистра RCRx

Номер бита	Условное обозначение	Назначение
0:2	WLEN	Это поле используется для выбора длины принимаемого слова. Принимаемое слово всегда поступает в младшие биты регистра LRBRx (или RRBRx). При этом, если реальная длина принимаемого слова больше установленного значения, то будут прочитаны только его старшие биты. Пример:

Номер бита	Условное обозначение	Назначение
		Допустим, что установленная длина принимаемого слова – 16 бит. А по линии SDI поступает 32-х разрядное слово 0x12345678. В этом случае, через регистр LRBR0 (или RRBR0) будет прочитано слово 0x00001234. 000: зарезервировано. 001: 12 бит. 010: 16 бит. 011: 20 бит. 100: 24 бита. 101: 32 бита. Приемник должен быть выключен перед любыми изменениями этого значения. (RERx = 0).
1:31	-	Резерв

31.2.15 Регистр TCRx (for x=0; x <= I2C_TX_CHANNELS-1)

Регистр конфигурации передатчика x.

Формат регистра TCRx приведен в Таблица 31.16.

Таблица 31.16. Формат регистра TCRx

Номер бита	Условное обозначение	Назначение
0:2	WLEN	Длина передаваемого слова. Для передачи всегда используются старшие биты 32-х разрядного регистра LTHRx (или RTHRx). 000: зарезервировано 001: 12 бит. 010: 16 бит. 011: 20 бит. 100: 24 бита. 101: 32 бита. Передатчик x должен быть выключен перед любыми изменениями этого значения. (TERx = 0).
1:31	-	Резерв

31.2.16 Регистр ISRx (for x=0; x <= I2C_TX_CHANNELS-1)

Регистр статуса прерываний канала x.

Формат регистра ISRx приведен в Таблица 31.17.

Таблица 31.17. Формат регистра ISRx

Номер бита	Условное обозначение	Назначение
0	RXDA	При x = 0, этот бит содержит статус прерывания по наличию данных в приемнике. 1: Превышен порог заполнения FIFO приемника 0: Не превышен порог заполнения FIFO приемника

Номер бита	Условное обозначение	Назначение
1	RXFO	При x = 0, этот бит содержит статус прерывания по переполнению приемника. 0: Не произошло потери данных 1: Произошла потеря данных. В случае поступления данных по каналу SDI0, при заполненном буфере приемника, произойдет потеря этих данных. Данные в FIFO приемника не будут перезаписаны.
2:3	-	Резерв
4	TXFE	Статус прерывания по пустому FIFO передатчика x. 1: Число записей в FIFO передатчика меньше порога заполнения. 0: Число записей в FIFO передатчика не меньше порога заполнения.
5	TXFO	Статус прерывания по переполнению передатчика x. 0: Не произошло потери данных 1: Произошла потеря данных В случае поступления данных в FIFO передатчика x, при заполненном буфере передатчика, произойдет потеря этих данных. Данные в FIFO не будут перезаписаны
6:31	-	Резерв

31.2.17 Регистр IMRx (for x=0; x <= I2C_TX_CHANNELS-1)

Регистр маскирования прерываний в канале x, где x = 0,1,2,3.

Формат регистра IMRx приведен в Таблица 31.18.

Таблица 31.18. Формат регистра IMRx

Номер бита	Условное обозначение	Назначение
0	RXDAM	При x = 0, этот бит содержит маску прерывания по наличию данных в приемнике. 1: Прерывание маскировано 0: Прерывание не маскировано
1	RXFOM	При x = 0, этот эти содержит маску прерывания по переполнению приемника. 1: Прерывание маскировано 0: Прерывание не маскировано
2:3	-	Резерв
4	TXFEM	Маска прерывания по пустому FIFO передатчика x. 1: Прерывание маскировано 0: Прерывание не маскировано
5	TXFOM	Маска прерывания по переполнению передатчика x. 1: Прерывание маскировано 0: Прерывание не маскировано
6:31	-	Резерв

31.2.18 Регистр RORx (for x=0; x <= I2C_RX_CHANNELS-1)

Регистр переполнения FIFO приемника.

Формат регистра RORx приведен в Таблица 31.19.

Таблица 31.19. Формат регистра RORx

Номер бита	Условное обозначение	Назначение
0	RXCHO	Чтение этого бита сбрасывает прерывание по переполнению приемника. 0: Не произошло потери данных 1: Произошла потеря данных.
1:31	-	Резерв

31.2.19 Регистр TORx (for x=0; x <= I2C_TX_CHANNELS-1)

Регистр переполнения FIFO передатчика x.

Формат регистра TORx приведен в Таблица 31.20.

Таблица 31.20. Формат регистра TORx

Номер бита	Условное обозначение	Назначение
0	TXCHO	Чтение этого бита сбрасывает прерывание по переполнению передатчика x. 0: Не произошло потери данных 1: Произошла потеря данных.
1:31	-	Резерв

31.2.20 Регистр RFCRx (for x=0; x <= I2C_RX_CHANNELS-1)

Регистр конфигурации FIFO приемника.

Формат регистра RFCRx приведен в Таблица 31.21.

Таблица 31.21. Формат регистра RFCRx

Номер бита	Условное обозначение	Назначение
1:3	RXCHDT	Это поле содержит пороговое значение заполнения FIFO приемника, при достижении которого устанавливается прерывание по наличию данных в приемнике. Приемник должен быть выключен перед любыми изменениями этого значения. (RER0 = 0).
4:31	-	Резерв

31.2.21 Регистр TFCRx (for x=0; x <= I2C_TX_CHANNELS-1)

Регистр конфигурации FIFO передатчика x.

Формат регистра TFCRx приведен в Таблица 31.22.

Таблица 31.22. Формат регистра TFCRx

Номер бита	Условное обозначение	Назначение
0:3	TXCHET	Это поле содержит пороговое значение заполнения FIFO передатчика x. Если число записей в FIFO передатчика x меньше этого значения, то устанавливается прерывание по пустому FIFO передатчика x.

Номер бита	Условное обозначение	Назначение
		Передатчик должен быть выключен перед любыми изменениями этого значения. (TERx = 0).
4:31	-	Резерв

31.2.22 Регистр RFFx (for x=0; x <= I2C_RX_CHANNELS-1)

Регистр сброса FIFO приемника.

Формат регистра RFFx приведен в Таблица 31.23.

Таблица 31.23. Формат регистра RFFx

Номер бита	Условное обозначение	Назначение
0	RXCHFR	Запись единицы в этот регистр приводит к стиранию данных в FIFO приемника. Перед записью в этот регистр необходимо выключить приемник (RER0[0]=0) либо блок приема (IRER[0]=0).
1:31	-	Резерв

31.2.23 Регистр TFFx (for x=0; x <= I2C_TX_CHANNELS-1)

Регистр сброса FIFO передатчика x. Где x = 0,1,2,3.

Формат регистра TFFx приведен в Таблица 31.24.

Таблица 31.24. Формат регистра TFFx

Номер бита	Условное обозначение	Назначение
0	TXCHFR	Запись единицы в этот регистр приводит к стиранию данных в FIFO передатчика x. Перед записью в этот регистр необходимо выключить передатчик x (TERx[0] = 0) либо блок передачи (ITER[0]=0).
1:31	-	Резерв

31.2.24 Регистр RXDMA

Регистр DMA приема.

Этот регистр позволяет цикличное чтение «левого» и «правого» слов данных из FIFO буфера приемника. После включения приемника, первое чтение этого регистра соответствует чтению регистра LRBR0, второе – RRBR0, третье – LRBR0, и т.д. Выключение приемника в момент, когда из FIFO приемника прочитано «левое» слово, но не прочитано «правое», невозможно.

Формат регистра RXDMA приведен в Таблица 31.25.

Таблица 31.25. Формат регистра RXDMA

Номер бита	Условное обозначение	Назначение

0:31	RXDMA	Регистр для цикличного чтения стереопар данных из приемника.

31.2.25 Регистр TXDMA

Регистр DMA передачи.

Этот регистр позволяет цикличную запись «левого» и «правого» слов данных в FIFO буферы всех включенных передатчиков. Так например, если включены нулевой и второй передатчики, то первая запись в этот регистр соответствует записи в регистра LTHR0, вторая – RTHR0, третья – LTHR2, четвертая – RTHR2, пятая – опять LTHR0 и т.д. Передатчики могут включаться и выключаться в процессе цикла записи в регистр TXDMA, однако, выключение приемника в момент, когда в него записано «левое» слово, но не записано «правое», невозможно.

Формат регистра TXDMA приведен в Таблица 31.26.

Таблица 31.26. Формат регистра TXDMA

Номер бита	Условное обозначение	Назначение
0:31	RXDMA	Регистр для цикличной записи стереопар данных в передатчики.

31.2.26 Регистр RTXDMA

Регистр сброса DMA передачи.

Этот регистр устанавливает в качестве текущего передатчика циклической DMA передачи включенный передатчик с младшим индексом (LTHR0, при TER0[0] = 1). При этом, запись в этот регистр не будет иметь эффекта в момент, когда в FIFO текущего передатчика записано «левое» слово, но не записано «правое».

Формат регистра TXDMA приведен в Таблица 31.27.

Таблица 31.27. Формат регистра RTXDMA

Номер бита	Условное обозначение	Назначение
0	RTXDMA	Запись единицы в этот бит установит в качестве текущего передатчика циклической DMA передачи включенный передатчик с младшим индексом.
1:31	-	Резерв

31.2.27 Регистр I2S_COMP_PARAM_2

Второй регистр параметров компонента.

Этот read-only регистр хранит информацию о конфигурации контроллера.

Формат регистра I2S_COMP_PARAM_2 приведен в Таблица 31.28.

Таблица 31.28. Формат регистра I2S_COMP_PARAM_2

Номер бита	Условное обозначение	Назначение
0:2	I2S_RX_WORDSIZE_0	Максимальный размер принимаемого слова: 0x0 = 12 бит 0x1 = 16 бит 0x2 = 20 бит 0x3 = 24 бит 0x4 = 32 бит 0x5-0x7 = Резерв
3:5	I2S_RX_WORDSIZE_1	
6	-	Резерв
7:9	I2S_RX_WORDSIZE_2	Максимальный размер принимаемого слова: 0x0 = 12 бит 0x1 = 16 бит 0x2 = 20 бит 0x3 = 24 бит 0x4 = 32 бит 0x5-0x7 = Резерв
10:12	I2S_RX_WORDSIZE_3	
13:31	-	Резерв

31.2.28 Регистр I2S_COMP_PARAM_1

Первый регистр параметров компонента.

Этот read-only регистр хранит информацию о конфигурации контроллера.

Формат регистра I2S_COMP_PARAM_1 приведен в Таблица 31.29.

Таблица 31.29. Формат регистра I2S_COMP_PARAM_1

Номер бита	Условное обозначение	Назначение
0:1	APB_DATA_WIDTH	Требуемая ширина шины APB для подключения к контроллеру: 0x0 = 8 бит 0x1 = 16 бит 0x2 = 32 бита 0x3 = резерв
2:3	I2S_FIFO_DEPTH_GLOBAL	Глубина FIFO – буферов всех каналов прима и передачи: 0x0 = 2 слова 0x1 = 4 слова 0x2 = 8 слов 0x3 = 16 слов
4	I2S_MODE_EN	Режим работы контроллера: 0x0 = Контроллер работает как Slave 0x1 = Контроллер работает как Master
5	I2S_TRANSMITTER_BLOCK	Наличие блока передачи: 0x0 = Контроллер не имеет блока передачи 0x1 = Контроллер имеет блок передачи
6	I2S_RECEIVER_BLOCK	Наличие блока приема: 0x0 = Контроллер не имеет блока приема 0x1 = Контроллер имеет блок приема

Номер бита	Условное обозначение	Назначение
7:8	I2S_RX_CHANNELS	Число каналов приема: 0x0 – 1 канал 0x1 – 2 канала 0x2 – 3 канала 0x3 – 4 канала
9:10	I2S_TX_CHANNELS	Число каналов передачи: 0x0 – 1 канал 0x1 – 2 канала 0x2 – 3 канала 0x3 – 4 канала
11:15	-	Резерв
16:18	I2S_TX_WORDSIZE_0	Максимальный размер передаваемого слова: 0x0 = 8 бит 0x1 = 16 бит 0x2 = 32 бита 0x3 = резерв
19:21	I2S_TX_WORDSIZE_1	
22:24	I2S_TX_WORDSIZE_2	
25:27	I2S_TX_WORDSIZE_3	
28:31	-	Резерв

31.2.29 Регистр I2S_COMP_VERSION

Регистр версии компонента.

Формат регистра I2S_COMP_VERSION приведен в Таблица 31.30.

Таблица 31.30. Формат регистра I2S_COMP_VERSION

Номер бита	Условное обозначение	Назначение
0:31	I2S_COMP_VERSION	Версия DesignWare компонента.

31.2.30 Регистр I2S_COMP_TYPE

Регистр типа компонента.

Формат регистра I2S_COMP_VERSION приведен в Таблица 31.31.

Таблица 31.31. Формат регистра I2S_COMP_TYPE

Номер бита	Условное обозначение	Назначение
0:31	I2S_COMP_TYPE	Тип DesignWare компонента.

31.2.31 Регистр RRXDMA

Регистр сброса DMA приема.

RXDMA может быть сброшен на самый нижний включенный канал через регистр RRXDMA. Регистр RRXDMA может быть записан на любом этапе цикла чтения RXDMA.

Формат регистра RRXDMA приведен в Таблица 31.32.

Таблица 31.32. Формат регистра RRXDMA

Номер бита	Условное обозначение	Назначение
31:1	RSVD_RRXDMA	Резерв
0	RRXDMA	Запись 1 сбрасывает регистр RXDMA в середине цикла.

31.3 Функционирование контроллера I²S

I²S шина может обрабатывать только передачу аудиоданных, субкодирование и управление осуществляется другим устройством, например I²C. Протокол I²S требует минимум трех проводов:

- Сигнал данных (sd);
- Тактовый сигнал фреймов (по словам) синхронизации (ws);
- Тактовый сигнал битовой синхронизации (sclk).

Также, I²S можно настроить так, чтобы было до четырех каналов для операций передачи и приема (10 проводов).

I²S можно настроить для работы в качестве мастера или слейва. В качестве мастера I²S инициализирует выбор слова (сигнал ws_out) и подает сигналы синхронизации (sclk_gate) и разрешения синхронизации (sclk_en). В качестве слейва I²S реагирует на внешние сигналы sclk и ws.

Независимо от того, настроен ли он, как мастер или слейв, внешний sclk и инвертированный sclk должны подаваться на устройство через входные сигналы sclk и sclk_n.

На Рисунок 31.1 показана блок-схема I²S.

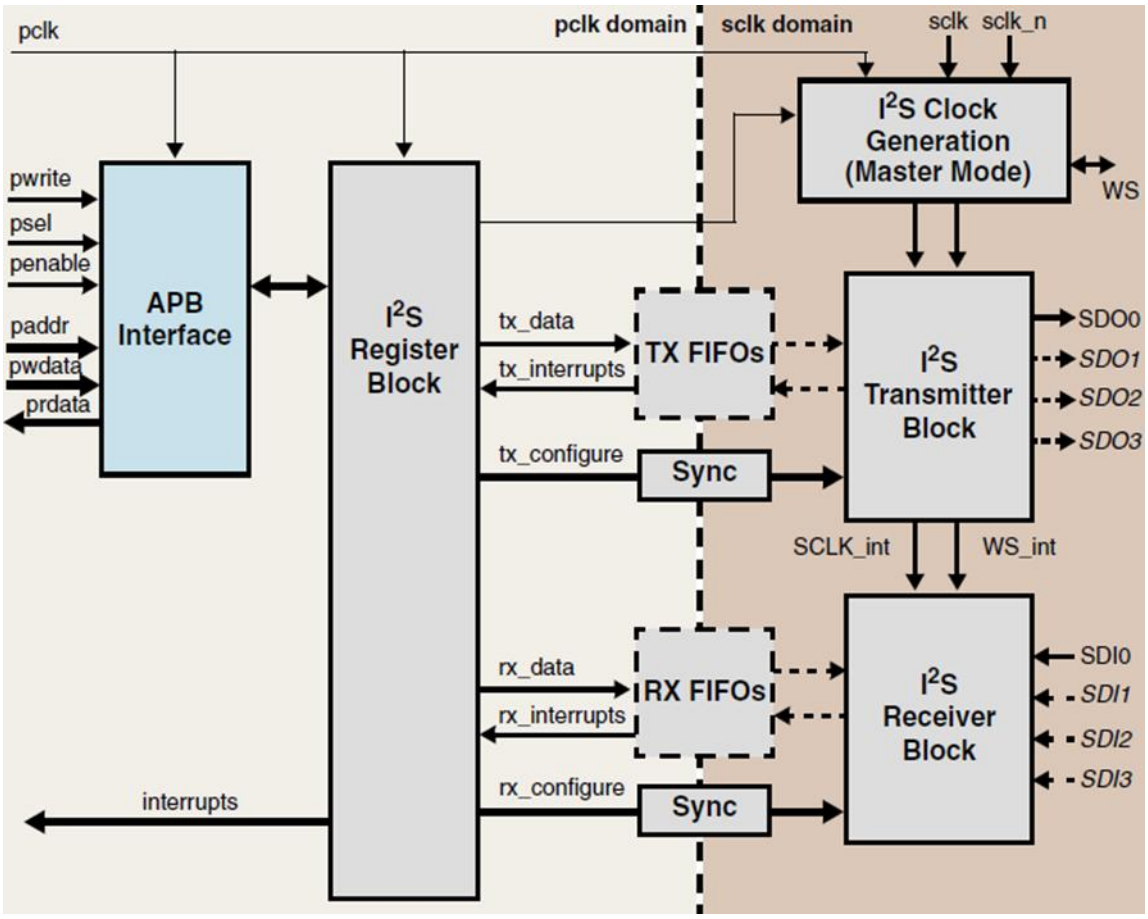


Рисунок 31.1. Блок-схема I²S

31.4 Включение и выключение контроллера I²S

Контроллер I²S должен быть включен перед любым обменом данными по шине I²S. Для включения контроллера необходимо записать 1 в бит IEN регистра IER. Для выключения контроллера необходимо записать 0 в бит IEN регистра IER.

После выключения контроллера произойдет следующее:

1. Все содержимое FIFO приемников и передатчика будет стерто.
2. Любые данные, находящиеся в процессе передачи или приема будут потеряны.
3. Будут выключены блоки приема и передачи.
4. Прекратится генерация внутренних сигналов управления выходной частотой SCLK_EN, SCLK_GATE и сигнала выбора слова WS_OUT.

Когда контроллер I²S включен, любой обмен данными всегда начинается с «левого» слова данных (WS=0). Завершение обмена всегда происходит на следующий такт после «правого» слова данных (WS=1).

При сбросе IER[0] устанавливается равным 0 (отключено).

31.5 I²S в качестве передатчика

I²S может быть сконфигурирован для поддержки до четырех каналов передачи. Эти каналы могут работать, как в режиме мастера, так и слейва. По умолчанию, I²S является слейвом. Стереопары данных, записанные в канал TX через шину APB, последовательно сдвигаются по sdo0, sdo1, sdo2, sdo3. Сдвиг выполняется относительно sclk и ws.

Реализация блока передатчик I²S и количество каналов TX определяется I2S_TRANSMITTER_BLOCK и I2S_TX_CHANNELS. По умолчанию I²S настроен на один канал передачи.

31.5.1 Включение блока передачи

Блок передачи I²S должен быть включен перед любой записью данных на шину I²S. Для включения блока передачи необходимо записать 1 в бит TXEN регистра ITER. Для выключения необходимо записать 0 в бит TXEN регистра ITER.

После выключения блока передачи произойдет следующее:

1. Будут выключены все передатчики блока передачи (все TER*x*[0] = 0).
2. Данные, передаваемые в данный момент, будут потеряны, и линии SDO будут удерживать постоянный низкий уровень.
3. Все данные в FIFO каждого передатчика сохранятся, также возможно будет осуществлять дальнейшую запись данных в FIFO каждого передатчика.
4. Все предыдущие настройки параметров передатчиков (длина слова, пороговое значение заполнения FIFO и т.д.) сохранятся.
5. Когда блок передачи выключен, возможно выполнение следующих операций:
   5.1. Настройка параметров передатчиков.
   5.2. Стирание данных во всех FIFO блока передачи. Для этого необходимо записать 1 в бит TXFFR[0].
   5.3. Стирание данных в FIFO конкретного передатчика x. Для этого необходимо записать 1 в бит TFF*x*[0]. Эту операцию можно проводить и при включенном блоке передаче, но выключенном передатчике *x*.

При сбросе значение ITER[0] устанавливается равным в 0 (отключено).

31.5.2 Включение канала передачи

Блок передачи включает в себя четыре независимых передатчика, каждый из которых может включаться и выключаться независимо от остальных. Включение и выключение передатчика *x* (где *x* – номер канала) управляется через бит TXCHEN*x* регистра TER*x* при включенном блоке передачи (ITER[0]=1).

При выключенном TX канале произойдет следующее:

1. Исходящие стереоданные теряются.
2. Выход канала удерживается на низком уровне.
3. Данные в TX FIFO сохраняются и в FIFO можно записывать.
4. Все предыдущие настройки канала TX сохраняются, также регистры могут быть перепрограммированы.

Когда канал TX выключен, возможно выполнение следующей операции:

- Стирание данных во всех FIFO блока передачи. Для этого необходимо записать 1 в бит TXFFR[0].

Когда канал TX включен, возможно выполнение следующей операции:

- Если в TX FIFO есть данные, то канал возобновляет передачу.

При сбросе значение TFFx[0] устанавливается равным в 1 (разрешено).

31.5.3 Канал передачи аудиоданных

Каждый канал TX изначально настроен на максимальное разрешение аудиоданных (I2S_TX_WORDSIZE_x). Канал TX может быть перепрограммирован во время работы на любое поддерживаемое разрешение аудиоданных, которое меньше, чем I2S_TX_WORDSIZE_x, при этом формат данных остается прежнем: старший бит данных передается первым.

При сбросе или если выбрано недопустимое разрешение, то канал передачи аудиоданных по умолчанию возвращается к начальной настройке параметра I2S_TX_WORDSIZE_x.

31.5.4 Канал передачи FIFO

У каждого канала передачи есть два FIFO для левых и правых стереоданных. FIFO можно настроить в соответствии параметром I2S_FIFO_DEPTH_GLOBAL.

Есть несколько способов, чтобы очистить буферы TX FIFO и сбросить указатели чтения/записи:

- При сбросе.
- Отключить I^2S (IER[0] = 0).
- Очистить блок передачи (TXFFR[0] = 1).
- Очистить отдельный канал передачи (TFFx[0] = 1).

Чтобы очистить блок передачи и отдельный канал FIFO необходимо отключить блок/канал передатчика.

I2S_TX_FIFO_THRE_x устанавливает пороговый уровень для TX FIFO (диапазон от 0 до I2S_TX_FIFO_x – 1). При достижении этого уровня генерируется прерывание. Данный уровень можно перепрограммировать во время работы, записав в биты TXCHET регистра TFCRx, отключив канал TX.

31.5.5 Прерывания канала передачи

Все прерывания в I^2S могут быть сконфигурированы параметром I2S_INTR_POL. Каждый канал генерирует два прерывания:

- TX FIFO Empty interrupt— утверждается при достижении порогового значения для TX FIFO. Когда это прерывание включено в систему ввода-вывода, оно появляется на выходах tx_emp_x_intr. Прерывание очищается путем записи данных в TX FIFO.

- Data Overrun interrupt — утверждается, если делается попытка записи в полный буфер TX FIFO. Когда это прерывание включено в систему ввода-вывода, оно появляется на выходах tx_or_x_intr. Прерывание очищается путем чтения TORx[0].

Статус прерывания может быть определен регистром ISRx. Бит TXFE[4] указывает состояние прерывания TX FIFO Empty, бит TXFO[5] — Data Overrun.

Оба прерывания можно замаскировать, записав 1 в биты TXFEM и TXFOM регистра IMRx. Это не позволяет прерываниям управлять своими выходными линиями, однако ISRx всегда показывает текущее состояние прерываний независимо от какой-либо маскировки.

Настройка I2S_INTERRUPT_SIGNALS влияет на то, включены ли эти два прерывания в интерфейсе I^2S. В Таблица 31.33 показаны конкретный сигнал прерывания, который должен отображаться при вводе/выводе в соответствии с настройкой I2S_INTERRUPT_SIGNALS.

Таблица 31.33. Сигналы прерывания канала TX

Interrupt	Signal on I/O
I2S_INTERRUPT_SIGNALS = True	
TX FIFO Empty	tx_emp_*x*_intr
Data Overrun	tx_or_*x*_intr
I2S_INTERRUPT_SIGNALS = False	
Combined interrupt signal	intr

31.5.6 Запись канала передачи

Пары стереоданных, которые должны быть переданы каналом TX, записываются в TX FIFO через регистры LTHRx и RTHRx. Все пары стереоданных должны быть записаны с помощью двухэтапного процесса:

1. Запись данных левого стерео в LTHRx.

2. Запись данных правого стерео в RTHRx.

Если I2S_TX_DMA = 1, то данные, передаваемые по каналам TX, записываются в TX FIFO через регистр TXDMA. Данные записываются циклически через все включенные каналы TX. После передачи стереопары компонент укажет на следующий включенный канал.

Если ширина шины APB меньше сконфигурированного или запрограммированного разрешения звука, то потребуется несколько операция записи для каждого стереослова.

31.6 I²S в качестве приемника

I²S может быть сконфигурирован для поддержки до четырех стереоканалов приема. Эти каналы могут работать, как в режиме мастера, так и слейва. По умолчанию, I²S является слейвом. Пары стереоданных поступают последовательно из линии ввода данных (sdi0, sdi1, sdi2, sdi3). Эти слова данных хранятся в RX FIFO до тех пор, пока они не будут считаны через шину APB. Прием выполняется относительно sclk и ws.

Реализация блока приемника I²S и количество каналов RX определяется I2S_RECEIVER_BLOCK и I2S_RX_CHANNELS. По умолчанию I²S настроен на один канал приема.

31.6.1 Включение принимающего блока

Блок приема контроллера I²S должен быть включен перед любым чтением данных с шины I²S. Для включения блока приема необходимо записать 1 в бит RXEN регистра IRER. Для выключения блока приема необходимо записать 0 в бит RXEN регистра IRER.

После выключения блока приема контроллера I²S произойдет следующее:

- приемник блока приема будет выключен (RER0[0]=0);
- данные, принимаемые в текущий момент, будут потеряны, и линия SDO будет удерживать постоянный низкий уровень;
- все данные в FIFO приемника сохранятся, также возможно будет осуществлять дальнейшее чтение данных из FIFO приемника;
- все предыдущие настройки параметров приемника (длина слова, пороговое значение заполнения FIFO и т.д.) сохранятся;
- когда блок приема выключен, возможно выполнение следующих операций:
  - настройка параметров приемника;
  - стирание данных из FIFO приемника. Для этого необходимо записать 1 в бит RXFFR[0], либо 1 в бит RFF0[0].

Блок приема состоит из одного приемника, который можно включать и выключать при включенном блоке приема.

При сбросе значение IRER[0] равно 0 (отключено).

31.6.2 Включение принимающего канала

Каждый канал RX имеет свое собственное включение/выключение, которое может быть установлено независимо от других каналов (RERx[0], где x – номер канала).

Когда канал RX отключен, происходит следующее:

- данные, принимаемые в текущий момент, будут потеряны;
- все данные в FIFO сохранятся, также возможно будет осуществлять дальнейшее чтение данных из FIFO;
- все предыдущие настройки параметров канала RX сохраняются, также допускаются дополнительные настройки.

Когда канал RX выключен, возможно выполнить следующие операции:

- стирание данных из RX FIFO канала. Для этого необходимо записать 1 в бит RFFx[0].

При сбросе RFFx[0] установить в 1 (включено).

31.6.3 Разрешение аудиоданных принимающего канала

Каждый канал RX изначально настроен на максимальное разрешение аудиоданных. Канал RX может быть запрограммирован во время работы на любое поддерживаемое разрешение аудиоданных, которое меньше, чем I2S_RX_WORDSIZE_x. Если разрешение принимаемых данных снижается, то LSB принятых данных помещается в LSB FIFO RX. Изменение разрешения программируется через биты WLEN регистров RCRx[3:0]. Перед любым изменением разрешения канала должен быть отключен.

Канал RX также поддерживает неизвестные разрешения данных. Если полученное слово превышает настроенное разрешение канала, младшие значащие биты игнорируются. Если полученное слово меньше настроенного/запрограммированного разрешения канала, младшие значащие биты дополняются нулями.

При сбросе или если выбрано недопустимое разрешение, разрешение аудиоданных канала RX возвращается по умолчанию к начальной настройке параметра I2S_RX_WORDSIZE_x.

31.6.4 FIFO принимающего канала

Каждый принимающий канал имеет два FIFO для левых и правых стереоданных. Настройка FIFO осуществляется через параметр I2S_FIFO_DEPTH_GLOBAL. Ширина FIFO определяется сконфигурированным максимальным разрешением данных для канала (I2S_RX_WORDSIZE_x, где x – номер канала).

Можно очистить RX FIFO или сбросить указатель чтения/записи следующими способами:

- при сбросе;
- отключить I^2S (IER[0] = 0);
- отключить блок приема (RXFFR[0] = 1);
- отключить отдельный канал RX (RX[0] = 1).

Перед очисткой блока приемника или отдельных каналов необходимо отключить блок приемника или канал.

31.6.5 Прерывания принимающего канала

Каждый канал RX генерирует два прерывания:

- RX FIFO Data Available — прерывание устанавливается, когда достигает уровень триггера для RX FIFO. Если включено в ввод/вывод, то появляется на выходах rx_da_*x*_intr. Данное прерывание снимается путем чтения данных из RX FIFO до тех пор, пока уровень не опуститься ниже уровня триггера, доступного для канала. Прерывание показывается битом RXDA[0].

- Data Overrun — прерывание устанавливается, когда полученные данные пытаются записать в полное RX FIFO (сами данные теряются, в то время, как данные в FIFO сохраняются). Если включено в ввод/вывод, то появляется на выходах rx_or_*x*_intr. Данное прерывание снимается путем считывания RXCHO[0] регистра RORx . Прерывание показывается битом RXDA[1].

Оба прерывания могут быть замаскированы путем записи 1 в биты RDM и ROM регистра IMRx. Это не позволяет прерываниям управлять своими выходами, однако IRSx всегда показывает текущее состояние прерываний независимо от какой-либо маскировки.

Таблица 31.34. Сигналы прерывания канала RX

Interrupt	Signal on I/O
I2S_INTERRUPT_SIGNALS = True RX FIFO Data Available Data Overrun	 rx_da_*x*_intr rx_or_*x*_intr
I2S_INTERRUPT_SIGNALS = False Combined interrupt signal	 Intr

31.6.6 Чтение из принимающего канала

Стереопары данных, полученные RX каналом, записываются в левый и правый RX FIFO. Эти FIFO могут быть прочитаны через LRBRx и RRBRx. Все пары стереоданных должны быть считаны с помощью следующего двухэтапного процесса:

1. Считайте левые стереоданные из LRBRx.

2. Считайте правые стереоданные из RRBRx.

Когда RX DMA включен (I2S_RX_DMA = 1), данные могут быть считаны из RX FIFO через регистр RXDMA.

31.7 Блок управления тактовым сигналом SCLK

Для генерации сигналов SCLK и WS_OUT необходимо установить 1 в бит CLKEN регистра CER. Установка 0 в бит CLKEN регистра CER прекращает генерацию сигналов SCLKO и WS_OUT.

Когда бит CLKEN отключен, все входящие или исходящие данные теряются. Однако данные, уже имеющиеся в FIFO RX и TX, сохраняются.

Сигнал sclk_en, который может быть объединен с помощью AND с sclk, позволяет отключить тактирование, когда мастер отключен.

Любые изменения параметров генерации сигналов GATED_SCLK и WS_OUT должны производиться при выключенной генерации. Сигнал SCLK_EN обеспечивает выключение генерации сигнала GATED_SCLK при выключении контроллера I^2S.

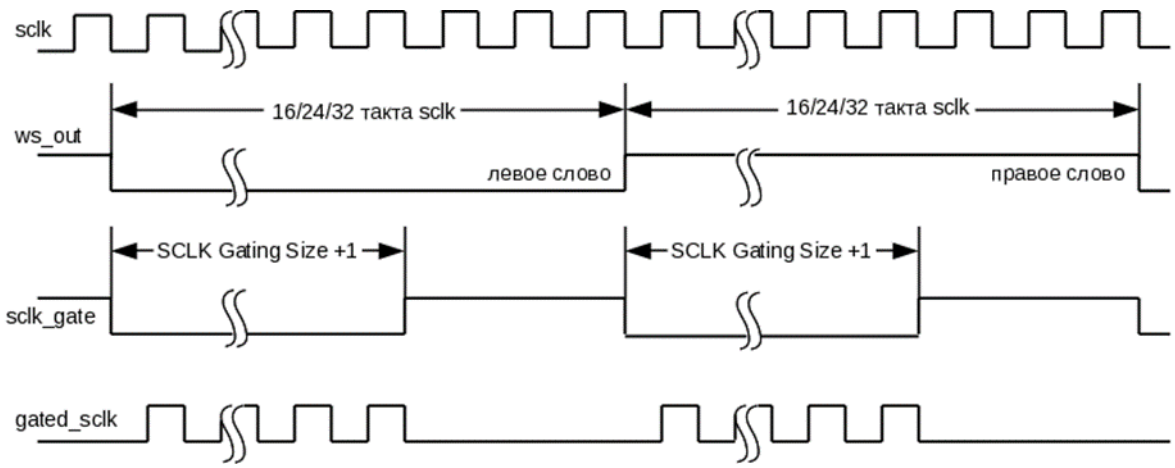


Рисунок 31.2. Формирование выходной частоты GATED_SCLK

32. КОНТРОЛЛЕР I²C (I2C)

32.1 Общая информация

Контроллер двухпроводного последовательного интерфейса (далее I2C) имеет следующие характеристики:

1. Микросхема содержит 5 контроллеров I2C.
2. Имеет три скоростных режима передачи данных с программируемой скоростью передачи внутри режима:
   - Standard-speed (0-100 Кбит/с)
   - Fast-speed (≤ 400 Кбит/с)
   - High-speed (≤ 3.4 Мбит/с). Только для I2C1-I2C4 контроллеров.
3. Поддерживает Multi-мастер режим (синхронизация тактовых частот, процедура арбитража при передаче данных).
4. Поддерживает 7-и и 10-и битную адресацию и возможность динамического переключения между ними.
5. Поддерживает режимы работы по прерыванию и по опросу.
6. Контроллеры I2C0-I2C3 имеют FIFO-буферы на 64 слова, I2C4 – на 8 слов.
7. Контроллеры I2C0-I2C3 поддерживают функцию GENERAL_CALL.
8. Контроллеры I2C0-I2C3 имеет интерфейс взаимодействия с DMA-контроллером.
9. Имеет настраиваемые параметры фильтрации помех.
10. Контролер I2C0 поддерживает протокол SMBUS.

32.2 Регистры

Перечень регистров приведен в Таблица 32.1.

Таблица 32.1. Перечень программно-доступных регистров

Смещение	Условное Обозначение	Название регистра	I2C0	I2C1-I2C3	I2C4	Тип доступа
0x0	IC_CON	Регистр управления	0x00000065	0x0000003F	0x0000007F	RW
0x4	IC_TAR	Регистр адреса абонента	0x00000055	0x00001055	0x00000055	RW
0x8	IC_SAR	Регистр slave-адреса.	0x00000055	0x00000055	0x00000055	RW
0xc	IC_HS_MADDR	Регистр кода адреса мастера для high speed режима.	-	0x00000001	0x00000001	RW
0x10	IC_DATA_CMD	Регистр управления передачей.	0x00000000	0x00000000	0x00000000	RW

Смещение	Условное Обозначение	Название регистра	I2C0	I2C1-I2C3	I2C4	Тип доступа
0x14	IC_SS_SCL_HCNT	Старший регистр счетчика предделителя частоты для standard-speed режима.	0x00000320	0x000001BD	0x00000190	RW
0x18	IC_SS_SCL_LCNT	Младший регистр счетчика предделителя частоты для standard-speed режима.	0x000003AC	0x0000020B	0x000001D6	RW
0x1c	IC_FS_SCL_HCNT	Старший регистр счетчика предделителя частоты для fast-speed режима.	0x00000078	0x00000043	0x0000003C	RW
0x20	IC_FS_SCL_LCNT	Младший регистр счетчика предделителя частоты для fast-speed режима.	0x00000104	0x00000091	0x00000082	RW
0x24	IC_HS_SCL_HCNT	Старший регистр счетчика предделителя частоты для high-speed режима.	-	0x00000007	0x00000006	RW
0x28	IC_HS_SCL_LCNT	Младший регистр счетчика предделителя частоты для high-speed режима.	-	0x00000012	0x00000010	RW
0x2c	IC_INTR_STAT	Регистр статуса прерывания.	0x00000000	0x00000000	0x00000000	RO
0x30	IC_INTR_MASK	Регистр маски прерывания	0x000048FF	0x000008FF	0x000008FF	RW
0x34	IC_RAW_INTR_STAT	Регистр статуса необработанного прерывания	0x00000000	0x00000000	0x00000000	RO
0x38	IC_RX_TL	Регистр порога заполнения FIFO приемника.	0x00000000	0x00000000	0x00000000	RW

Смещение	Условное Обозначение	Название регистра	I2C0	I2C1-I2C3	I2C4	Тип доступа
0x3c	IC_TX_TL	Регистр порога заполнения FIFO передатчика.	0x00000000	0x00000000	0x00000000	RW
0x40	IC_CLR_INTR	Регистр сброса прерываний.	0x00000000	0x00000000	0x00000000	RO
0x44	IC_CLR_RX_UNDER	Регистр сброса прерывания RX_UNDER	0x00000000	0x00000000	0x00000000	RO
0x48	IC_CLR_RX_OVER	Регистр сброса прерывания RX_OVER	0x00000000	0x00000000	0x00000000	RO
0x4c	IC_CLR_TX_OVER	Регистр сброса прерывания TX_OVER	0x00000000	0x00000000	0x00000000	RO
0x50	IC_CLR_RD_REQ	Регистр сброса прерывания RD_REQ	0x00000000	0x00000000	0x00000000	RO
0x54	IC_CLR_TX_ABRT	Регистр сброса прерывания TX_ABR	0x00000000	0x00000000	0x00000000	RO
0x58	IC_CLR_RX_DONE	Регистр сброса прерывания RX_DONE	0x00000000	0x00000000	0x00000000	RO
0x5c	IC_CLR_ACTIVITY	Регистр сброса прерывания ACTIVITY	0x00000000	0x00000000	0x00000000	RO
0x60	IC_CLR_STOP_DET	Регистр сброса прерывания STOP_DET	0x00000000	0x00000000	0x00000000	RO
0x64	IC_CLR_START_DET	Регистр сброса прерывания START_DET	0x00000000	0x00000000	0x00000000	RO
0x68	IC_CLR_GEN_CALL	Регистр сброса прерывания GEN_CALL	0x00000000	0x00000000	0x00000000	RO
0x6c	IC_ENABLE	Регистр включения шины.	0x00000000	0x00000000	0x00000000	RW
0x70	IC_STATUS	Регистр статуса шины.	0x00000006	0x00000006	0x00000006	RO
0x74	IC_TXFLR	Регистр уровня FIFO передатчика	0x00000000	0x00000000	0x00000000	RO

Смещение	Условное Обозначение	Название регистра	I2C0	I2C1-I2C3	I2C4	Тип доступа
0x78	IC_RXFLR	Регистр уровня FIFO приемника	0x00000000	0x00000000	0x00000000	RO
0x7c	IC_SDA_HOLD	Регистр времени удержания SDA.	0x00000001	0x00000001	0x00000001	RW
0x80	IC_TX_ABRT_SOURCE	Регистр статуса обрыва передачи.	0x00000000	0x00000000	0x00000000	RO
0x88	IC_DMA_CR	Регистр контроля DMA интерфейса.	0x00000000	0x00000000	0x00000064	RW
0x8c	IC_DMA_TDLR	Регистр передачи данных через DMA интерфейс.	0x00000000	0x00000000	0x00000001	RW
0x90	IC_DMA_RDLR	Регистр приема данных через DMA интерфейс.	0x00000000	0x00000000	0x00000000	RW
0x94	IC_SDA_SETUP	Регистр установки SDA	0x00000064	0x00000064	0x00000005	RW
0x98	IC_ACK_GENERAL_CALL	Регистр вызова общего ACK	0x00000001	0x00000001	0x00000001	RW
0x9c	IC_ENABLE_STATUS	Регистр статуса включения	0x00000000	0x00000000	0x00000000	RO
0xa0	IC_FS_SPKLEN	Регистр управления фильтрацией для standard-speed и fast-speed режимов	0x0000000A	0x00000006	0x00000000	RW
0xa4	IC_HS_SPKLEN	Регистр управления фильтрацией для high-speed режима.	-	0x00000002	0x00000001	RW
0xa8	IC_CLR_RESTART_DET	Регистр сброса прерывания RESTART_DET	0x00000000	0x00000000	-	RO
0xac	IC_SCL_STUCK_AT_LOW_TIMEOUT	Регистр таймаута по удержанию низкого уровня линии SCL.	0xFFFFFFFF	-	-	RW

Смещение	Условное Обозначение	Название регистра	I2C0	I2C1-I2C3	I2C4	Тип доступа
0xb0	IC_SDA_STUCK_AT_LOW_TIMEOUT	Регистр таймаута по удержанию низкого уровня линии SDA.	0xFFFFFFFF	-	-	RW
0xb4	IC_CLR_SCL_STUCK_DET	Регистр сброса прерывания по таймауту по удержанию низкого уровня линии SCL	0x00000000	-	-	RO
0xbc	IC_SMBUS_CLK_LOW_SEXT	SMBus Slave Clock Extend Timeout register Регистр таймаута частоты	0xFFFFFFFF	-	-	RW
0xc0	IC_SMBUS_CLK_LOW_MEXT	SMBus Master Clock Extend Timeout register	0xFFFFFFFF	-	-	RW
0xc4	IC_SMBUS_THIGH_MAX_IDLE_COUNT	Регистр максимального значения для счетчика THigh Bus-idle.	0x0000FFFF	-	-	RW
0xc8	IC_SMBUS_INTR_STAT	Регистр статуса прерывания SMBUS	0x00000000	-	-	RO
0xcc	IC_SMBUS_INTR_MASK	Регистр маски прерывания SMBUS.	0x000007FF	-	-	RW
0xd0	IC_SMBUS_RAW_INTR_STAT	Регистр статуса немаскированного прерывания SMBUS.	0x00000000	-	-	RO
0xd4	IC_CLR_SMBUS_INTR	Регистр сброса прерывания SMBUS	0x00000000	-	-	WO
0xd8	IC_OPTIONAL_SAR	Регистр дополнительного Slave - адреса	0x00000000	-	-	RW

Смещение	Условное Обозначение	Название регистра	I2C0	I2C1-I2C3	I2C4	Тип доступа
0xdc	IC_SMBUS_UDID_LSB	Регистр SMBUS ARP UDID LSB	0xFFFFFFFF	-	-	RW
0xf4	IC_COMP_PARAM_1	Component Parameter Register 1	0x003F3FEA	0x001F1FEE	0x000707AE	RO
0xf8	IC_COMP_VERSION	I2C Component Version Register	0x3230312A	0x3230312A	0x3230312A	RO
0xfc	IC_COMP_TYPE	I2C Component Type Register	0x44570140	0x44570140	0x44570140	RO

32.2.1 Регистр IC_CON

Смещение:0x0

Регистр управления.

Регистр доступен для записи, только при выключенном контроллере I^2C (IC_ENABLE=0).

Формат регистра IC_CON приведен в Таблица 32.2:

Таблица 32.2. Поля регистра IC_CON

Разряды	Название	Описание	Сброс
31:20	-	Резерв	0x0
19	SMBUS_PERSISTENT_SLV_ADDR_EN	Бит управления контроллером режиме Slave как постоянным или непостоянным Slave'ом. Если режиме slave не постоянен, slave-контроллер сбрасывает флаг валидности адреса для команд General и Directed Reset ARP, иначе флаг валидности адреса всегда 1.	0x0
18	SMBUS_ARP_EN	Этот бит контролирует будет ли включена логика Address Resolution в режиме SMBus Slave. Если бит установлен в 1, контроллер будет декодировать и отвечать на команды Address Resolution протокола. Контроллер в режиме slave также включает генерацию/валидацию PEC байта для команд Adress Resolution протокола.	0x0
17	SMBUS_SLAVE_QUICK_EN	Бит включения QUICK-команд для SMBus Slave. 0: Контроллер в режиме SMBus-Slave 1: Контроллер в режиме SMBus-Slave принимает только Quick-команды.	0x0
16	OPTIONAL_SAR_CTRL	Бит разрешения использования IC_OPTIONAL_SAR регистра. Бит управления использованием регистра IC_OPTIONAL_SAR. Регистр IC_OPTIONAL_SAR используется в качестве дополнительного Slave-адреса.	0x0

Разряды	Название	Описание	Сброс
		Пользователь должен устновить валидный адрес в регистр IC_OPTIONA_SAR перед записью 1 в это поле.	
15:12	-	Резерв	0x0
11	BUS_CLEAR_FEATURE_CTRL	В режиме Master: 0: Возможность Bus Clear выключена 1: Возможность Bus Clear включена	0x0
10	STOP_DET_IF_MASTER_ACTIVE	Бит управления прерыванием STOP_DET для master-режима: 0: Прерывание STOP_DET устанавливается независимо от того, работает ли контроллер в режиме master или нет. 1: Прерывание STOP_DET устанавливается только когда контроллер работает в режиме master.	0x0
9	RX_FIFO_FULL_HLD_CTRL	Этот бит контролирует, будет ли контроллер удерживать шину, при полном Rx FIFO. 0: Произойдет переполнение Rx FIFO. 1: Контроллер будет удерживать шину при полном Rx FIFO	0x0
8	TX_EMPTY_CTRL	Бит управляет прерыванием TX_EMPTY, как описано в IC_RAW_INTR_STAT регистре.	0x0
7	STOP_DET_IFADDRESSED	Бит управления прерыванием STOP_DET для slave-режима. 0: Прерывание STOP_DET устанавливается при возникновении STOP-последовательности. 1: Прерывание STOP_DET устанавливается при возникновении STOP-последовательности. По адресу SAR	0x0
6	IC_SLAVE_DISABLE	Этот бит управляет выключением режима slave для контроллера. Если этот бит установлен, контроллер функционирует только как master 0: slave включен 1: slave выключен Перед установкой этого бита в 0, нужно убедиться, что бит 0 также установлен в 0.	0x1
5	IC_RESTART_EN	Бит определяет, может ли контроллер, работающий в режиме master, отправить последовательность RESTART. 0: не может отправить RESTART. 1: может отправить RESTART. Если IC_RESTART_EN=0, контроллер не может выполнять следующие функции: • Отправлять START-байт. • Выполнять любые обмены в режиме high-speed. • Изменять направление обмена в режиме комбинированного обмена.	0x1

Разряды	Название	Описание	Сброс
		• Выполнять операции чтения с 10-и битным адресом. В этом случае, вместо последовательности RESTART, будут отправлены последовательности STOP - START, что приведет к обрыву комбинированной передачи. При этом установится бит TX_ABRT (бит 6 регистра IC_RAW_INTR_STAT)	
4	IC_10BITADDR_MASTER_rd_only	Только для чтения. Бит отображает тип текущего адреса удаленного slave-устройства, к которому обращается контроллер, работающий в режиме master: 0: 7-и битный адрес 1: 10-и битный адрес Управление типом адреса производится битом IC_TAR(12).	0x0
3	IC_10BITADDR_SLAVE	Когда контроллер работает в режиме slave, этот бит выбирает тип адреса контроллера (7-и или 10-и битный адрес). 0: 7-и битный адрес. Контроллер игнорирует обращения по 10-и битным адресам, при этом учитываются только 7 младших битов регистра IC_SAR. 1: 10-и битный адрес. Контроллер отвечает только на 10-и битный адрес, который соответствует значению регистра IC_SAR.	0x0
2:1	SPEED	Режим скорости передачи данных. Настройка только для контроллера, работающего в режиме master. 1: Standard-speed (0 to 100 Кбит/с) 2: Fast-speed (≤ 400 Кбит/с) 3: High-speed (≤ 3.4 Мбит/с)	0x2
0	MASTER_MODE	Этот бит управляет включением режима master. 0: Режим master выключен. 1: Режим master включен. Перед установкой этого бита в 1, нужно убедиться, что бит 6 также установлен в 1.	0x1

32.2.2 Регистр IC_TAR

Смещение:0x4

I2C Target Address Register

Формат регистра IC_TAR приведен в Таблица 32.3:

Таблица 32.3. Поля регистра IC_TAR

Разряды	Название	Описание	Сброс
31:17	Резерв	-	0x0

Разряды	Название	Описание	Сброс
16	SMBUS_QUICK_CMD	Если бит 11 (SPECIAL) установлен в 1, то этот бит указывает будет ли выполняться контроллером Quick-команда.	0x0
15:13	-	Резерв	0x0
12	IC_10BITADDR_MASTER	Этот бит контролирует тип адреса удаленного slave-устройства. 0: 7-и битный адрес. 1: 10-и битная адресация	0x0
11	SPECIAL	Этот бит контролирует, будет ли происходить передача General Call, либо START-байт, или контроллер будет работать в обычном режиме. 0: Значение бита GC_OR_START игнорируется и контроллер будет в обычном режиме работать по адресу IC_TAR. 1: Будет выполнена специальная команда, в зависимости от значения бита GC_OR_START	0x0
10	GC_OR_START	Если бит SPECIAL установлен в 1, то бит GC_OR_START выбирает между выполнением General Call или Start-байта. 0: General Call. После выполнения этой команды, из всех транзакций становится возможна только передача данных по адресу General Call. При попытке чтения с шины произойдет установка бита TX_ABRT (IC_RAW_INTR_STAT[6]). Контроллер будет выполнять передачи по адресу General Call, до тех пор, пока не будет сброшен бит SPECIAL (IC_TAR[11]). 1: START-байт.	0x0
9:0	IC_TAR	Адрес удаленного slave-устройства, для любых транзакций контроллера, работающего в режиме master. Когда происходит передача General Call, то содержимое этих битов игнорируется.	0x55

32.2.3 Регистр IC_SAR

Смещение:0x8

I2C Slave Address Register

Формат регистра IC_SAR приведен в Таблица 32.4:

Таблица 32.4. Поля регистра IC_SAR

Разряды	Название	Описание	Сброс
31:10	RSVD_IC_SAR	IC_SAR Reserved bits - Read Only	0x0
9:0	IC_SAR	Регистр хранит адрес контроллера, работающего в режиме slave. Если установлен 7-и битный формат адреса (IC_CON=0), то используются только биты IC_SAR[6:0] Этот регистр доступен по записи только при выключенном контроллере (IC_ENABLE=0)	0x55

32.2.4 Регистр IC_HS_MADDR

Смещение:0xc

I2C High Speed Master Mode Code Address Register

Формат регистра IC_HS_MADDR приведен в Таблица 32.5:

Таблица 32.5. Поля регистра IC_HS_MADDR

Разряды	Название	Описание	Сброс
31:3	RSVD_IC_HS_MAR	IC_HS_MAR Reserved bits - Read Only	0x0
2:0	IC_HS_MAR	Эти биты содержат значение master-кода для high-speed режима. Каждое master -устройство имеет уникальный master -код вида (00001xxx). На одной шине может располагаться до 8-и master -устройств. Этот регистр может быть доступен для записи, только когда контроллер выключен (IC_ENABLE=0).	0x1

32.2.5 Регистр IC_DATA_CMD

Смещение:0x10

Через этот регистр происходит запись из CPU в буфер передатчика (TX FIFO) и чтение в CPU из буфера приемника (RX FIFO).

Формат регистра IC_DATA_CMD приведен в Таблица 32.6:

Таблица 32.6. Поля регистра IC_DATA_CMD

Разряды	Название	Описание	Сброс
31:12	-	Резерв	0x0
11	FIRST_DATA_BYTE	Индикатор приема первого байта данных после фазы адреса для режимов Master-приемник и Slave-приемник.	0x0
10	RESTART	Бит контролирует будет ли отправлена последовательность RESTERT перед отправлением или приемом байта. 1: Перед отправлением или приемом данных (в зависимости от бита CMD), независимо от того, менялось или нет направление передачи данных, будет отправлена RESTART-последовательность 0: Последовательность RESTART будет отправлена только в случае изменения направления передачи.	0x0
9	STOP	Бит управления генерацией STOP-последовательности после отправки или получения байта данных. 1: STOP-последовательность будет отправлена независимо от того пуст или нет Tx FIFO. Если Tx FIFO не пуст, мастер немедленно пытается начать новую передачу используя START-последовательность.	0x0

Разряды	Название	Описание	Сброс
		0: STOP-последовательность не будет отправлена независимо от того, пуст или нет Tx FIFO. Если Tx FIFO не пуст, master продолжает текущую передачу отправления/приема данных в соответствии со значением бита CMD. Если Tx FIFO пуст, мастер удерживает линию SCL на низком уровне и останавливает шину до тех пор, пока новая команда не появится в Tx FIFO.	
8	CMD	Бит направления передачи. Когда порт находится в режиме master, этот бит контролирует направление передачи: 1: Чтение. 0: Запись. Попытка выполнить операция чтения после General Call приведет к прерыванию TX_ABRT. Запись «0» в этот бит после прерывания RD_REQ также приведет к прерыванию TX_ABRT.	0x0
7:0	DAT	Поле данных. Это поле содержит передаваемые или принимаемые данные по шине I^2C.	0x0

32.2.6 Регистр IC_SS_SCL_HCNT

Смещение:0x14

Старший регистр счетчика предделителя частоты при работе модуля в standard-speed режиме.

Регистр доступен для записи только при выключенном контроллере (IC_ENABLE=0).

Формат регистра IC_SS_SCL_HCNT приведен в Таблица 32.7:

Таблица 32.7. Поля регистра IC_SS_SCL_HCNT

Разряды	Название	Описание	Сброс
31:16	-	Резерв	0x0
15:0	IC_SS_SCL_HCNT	Регистр устанавливает значение счетчика предделителя для высокого уровня сигнала SCL в standard-speed режиме. Минимальное значение - 6, максимальное - 65525.	0x320

32.2.7 Регистр IC_SS_SCL_LCNT

Смещение:0x18

Младший регистр счетчика предделителя частоты при работе модуля в standard-speed режиме.

Регистр доступен для записи только при выключенном контроллере (IC_ENABLE=0).

Формат регистра IC_SS_SCL_LCNT приведен в Таблица 32.8:

Таблица 32.8. Поля регистра IC_SS_SCL_LCNT

Разряды	Название	Описание	Сброс
31:16	-	Резерв	0x0
15:0	IC_SS_SCL_LCNT	Регистр устанавливает значение счетчика предделителя для низкого уровня сигнала SCL в standard-speed режиме. Минимальное значение - 8.	0x3ac

32.2.8 Регистр IC_FS_SCL_HCNT

Смещение:0x1c

Старший регистр счетчика предделителя частоты при работе модуля в Fast-speed режиме.

Регистр доступен для записи только при выключенном контроллере (IC_ENABLE=0).

Формат регистра IC_FS_SCL_HCNT приведен в Таблица 32.9:

Таблица 32.9. Поля регистра IC_FS_SCL_HCNT

Разряды	Название	Описание	Сброс
31:16	-	Резерв	0x0
15:0	IC_FS_SCL_HCNT	Регистр устанавливает значение счетчика предделителя для высокого уровня сигнала SCL в high-speed режиме. Минимальное значение - 6.	0x78

32.2.9 Регистр IC_FS_SCL_LCNT

Смещение:0x20

Младший регистр счетчика предделителя частоты при работе модуля в Fast-speed режиме.

Регистр доступен для записи только при выключенном контроллере (IC_ENABLE=0).

Формат регистра IC_FS_SCL_LCNT приведен в Таблица 32.10:

Таблица 32.10. Поля регистра IC_FS_SCL_LCNT

Разряды	Название	Описание	Сброс
31:16	-	Резерв	0x0
15:0	IC_FS_SCL_LCNT	Регистр устанавливает значение счетчика предделителя для низкого уровня сигнала SCL в fast-speed режиме. Минимальное значение - 8.	0x104

32.2.10 Регистр IC_HS_SCL_HCNT

Смещение:0x24

Старший регистр счетчика предделителя частоты при работе модуля в High-speed режиме.

Регистр доступен для записи только при выключенном контроллере (IC_ENABLE=0).

Формат регистра IC_HS_SCL_HCNT приведен в Таблица 32.11:

Таблица 32.11. Поля регистра IC_HS_SCL_HCNT

Разряды	Название	Описание	Сброс
31:16	-	Резерв	0x0
15:0	IC_HS_SCL_HCNT	Регистр устанавливает значение счетчика предделителя для высокого уровня сигнала SCL в high-speed режиме. Минимальное значение - 6.	0x7

32.2.11 Регистр IC_HS_SCL_LCNT

Смещение:0x28

Старший регистр счетчика предделителя частоты при работе модуля в High-speed режиме.

Регистр доступен для записи только при выключенном контроллере (IC_ENABLE=0).

Формат регистра IC_HS_SCL_LCNT приведен в Таблица 32.12:

Таблица 32.12. Поля регистра IC_HS_SCL_LCNT

Разряды	Название	Описание	Сброс
31:16	-	Резерв	0x0
15:0	IC_HS_SCL_LCNT	Регистр устанавливает значение счетчика предделителя для высокого уровня сигнала SCL в high-speed режиме. Минимальное значение - 6.	0x12

32.2.12 Регистр IC_INTR_STAT

Смещение:0x2c

Регистр статуса прерываний.

Каждый бит этого регистра имеет соответствующий маскирующий бит в регистре IC_INTR_MASK. Значения битов прерывания до маскирования доступны в регистре IC_RAW_INTR_STAT.

Формат регистра IC_INTR_STAT приведен в Таблица 32.13:

Таблица 32.13. Поля регистра IC_INTR_STAT

Разряды	Название	Описание	Сброс
31:15	-	Резерв	0x0

Разряды	Название	Описание	Сброс
14	R_SCL_STUCK_AT_LOW	См. Регистр IC_RAW_INTR_STAT.	0x0
13	R_MASTER_ON_HOLD		0x0
12	R_RESTART_DET		0x0
11	R_GEN_CALL		0x0
10	R_START_DET		0x0
9	R_STOP_DET		0x0
8	R_ACTIVITY		0x0
7	R_RX_DONE		0x0
6	R_TX_ABRT		0x0
5	R_RD_REQ		0x0
4	R_TX_EMPTY		0x0
3	R_TX_OVER		0x0
2	R_RX_FULL		0x0
1	R_RX_OVER		0x0
0	R_RX_UNDER		0x0

32.2.13 Регистр IC_INTR_MASK

Смещение:0x30

Регистр маскирования прерываний.

Биты этого регистра маскируют соответствующие биты регистра статуса прерываний. Значение 0 - маскирует прерывание, значение 1 - не маскирует.

Формат регистра IC_INTR_MASK приведен в Таблица 32.14:

Таблица 32.14. Поля регистра IC_INTR_MASK

Разряды	Название	Описание	Сброс
31:15	RSVD_IC_INTR_STAT	IC_INTR_STAT Reserved bits - Read Only	0x0
14	M_SCL_STUCK_AT_LOW	Эти биты маскируют прерывания, соответствующие битам регистра IC_INTR_STAT.	0x1
13	M_MASTER_ON_HOLD		0x0
12	M_RESTART_DET		0x0
11	M_GEN_CALL		0x1
10	M_START_DET		0x0
9	M_STOP_DET		0x0
8	M_ACTIVITY		0x0
7	M_RX_DONE		0x1
6	M_TX_ABRT		0x1
5	M_RD_REQ		0x1
4	M_TX_EMPTY		0x1
3	M_TX_OVER		0x1
2	M_RX_FULL		0x1
1	M_RX_OVER		0x1
0	M_RX_UNDER		0x1

32.2.14 Регистр IC_RAW_INTR_STAT

Смещение:0x34

Регистр статуса немаскированных прерываний.

При выключении контроллера, биты регистра остаются активными до тех, пока контроллер не перейдет в состояние IDLE.

Формат регистра IC_RAW_INTR_STAT приведен в Таблица 32.15:

Таблица 32.15. Поля регистра IC_RAW_INTR_STAT

Разряды	Название	Описание	Сброс
31:15	-	Резерв	0x0
14	SCL_STUCK_AT_LOW	Бит устанавливается в 1 когда SCL линия удерживается в низком уровне IC_SCL_STUCK_LOW_TIMEOUT тактов.	0 x0
13	MASTER_ON_HOLD	Бит устанавливается в 1 когда контроллер в режиме Master удерживает шину при пустом Tx FIFO.	0x0
12	RESTART_DET	Бит устанавливается в 1, когда на шине I2C возникает RESTART последовательность, контроллер находится в режиме Slave и адрес транзакции соответствует адресу контроллера.	0x0
11	GEN_CALL	Этот бит устанавливается в 1, если получен General Call адрес и отправлено подтверждение. Сбрасывается с выключением контроллера или чтением регистра IC_CLR_GEN_CALL.	0x0
10	START_DET	Этот бит устанавливается в 1, если на шине возникает START или RESTART состояние, независимо от того работает ли контроллер в режиме slave или master.	0x0
9	STOP_DET	Этот бит устанавливается в 1, если на шине возникает состояние STOP, независимо от того работает ли контроллер в режиме slave или master.	0x0
8	ACTIVITY	Установка этого бита в 1 говорит о том, что контроллер проявил какую-либо активность. Сброс прерывания возможен одним из четырех способов: • Выключение контроллера I^2C • Чтение регистра IC_CLR_ACTIVITY • Чтение регистра IC_CLR_INTR Системный сброс.	0x0
7	RX_DONE	Когда контроллер работает в режиме slave-передатчика, этот бит устанавливается в 1, если мастер не подтверждает передачу байта.	0x0
6	TX_ABRT	Этот бит устанавливается в 1, если модуль работает в режиме передатчика и не может произвести передачу. Когда этот бит устанавливается в 1, регистр IC_TX_ABRT_SOURCE отображает причину обрыва передачи.	0x0
5	RD_REQ	Бит устанавливается в 1, когда модуль работает в режиме slave, а текущий master шины запрашивает чтение данных из модуля. Контроллер удерживает состояние ожидания (SCL=0), пока прерывание обрабатывается. Процессор должен ответить на это прерывание и начать выдавать запрашиваемые данные в IC_DATA_CMD регистр. Этот бит установится в 0 после чтения регистра IC_CLR_RD_REQ.	0x0

Разряды	Название	Описание	Сброс
4	TX_EMPTY	Бит устанавливается в 1, когда кол-во данных в FIFO передатчика снижается до уровня, обозначенного в IC_TX_TL.	0x0
3	TX_OVER	Бит устанавливается в 1, во время передачи, если процессор пытается записать данные в IC_DATA_CMD при заполненном FIFO передатчика.	0x0
2	RX_FULL	Бит устанавливается в 1, когда буфер заполняется до уровня, обозначенного в регистре IC_RX_TL. Он автоматически сбрасывается, как только кол-во данных в буфере становится меньше этого уровня.	0x0
1	RX_OVER	Бит устанавливается в 1, если буфер приемника заполнен, но по шине приходит еще один байт данных. Контроллер выдаст сигнал подтверждения на шину, но все данные, полученные после заполнения FIFO приемника, будут потеряны.	0x0
0	RX_UNDER	Бит устанавливается в 1, когда пользователь пытается читать данные из пустого буфера приемника.	0x0

32.2.15 Регистр IC_RX_TL

Смещение:0x38

Регистр порога заполнения FIFO приемника.

Формат регистра IC_RX_TL приведен в Таблица 32.16:

Таблица 32.16. Поля регистра IC_RX_TL

Разряды	Название	Описание	Сброс
31:8	-	Резерв	0x0
7:0	RX_TL	Содержит пороговое значение заполнения FIFO приемника. Контролирует число записей в буфере, по достижению которого, формируется прерывание RX_FULL (2-й бит регистра IC_RAW_INTR_STAT). Значение изменяется от 0 до 255.	0x0

32.2.16 Регистр IC_TX_TL

Смещение:0x3c

Регистр порога заполнения FIFO передатчика.
Формат регистра IC_TX_TL приведен в Таблица 32.17:

Таблица 32.17. Поля регистра IC_TX_TL

Разряды	Название	Описание	Сброс
31:8	-	Резерв	0x0

Разряды	Название	Описание	Сброс
7:0	TX_TL	Содержит пороговое значение заполнения FIFO приемника. Контролирует число записей в буфере, по достижению которого, происходит прерывание TX_EMPTY (4-й бит регистра IC_RAW_INTR_STAT). Значение изменяется от 0 до 255.	0x0

32.2.17 Регистр IC_CLR_INTR

Смещение:0x40

Регистр сброса комбинированного и индивидуальных прерываний.

Формат регистра IC_CLR_INTR приведен в Таблица 32.18:

Таблица 32.18. Поля регистра IC_CLR_INTR

Разряды	Название	Описание	Сброс
31:1	-	Резерв	0x0
0	CLR_INTR	Чтение этого регистра сбрасывает комбинированное прерывание, все индивидуальные прерывания, регистр IC_TX_ABRT_SOURCE.	0x0

32.2.18 Регистр IC_CLR_RX_UNDER

Смещение:0x44

Регистр сброса прерывания RX_UNDER.

Формат регистра IC_CLR_RX_UNDER приведен в Таблица 32.19:

Таблица 32.19. Поля регистра IC_CLR_RX_UNDER

Разряды	Название	Описание	Сброс
31:1	-	Резерв	0x0
0	CLR_RX_UNDER	Чтение этого регистра сбрасывает прерывание RX_UNDER (бит 0 регистра IC_RAW_INTR_STAT)	0x0

32.2.19 Регистр IC_CLR_RX_OVER

Смещение:0x48

Регистр сброса прерывания RX_OVER.

Формат регистра IC_CLR_RX_OVER приведен в Таблица 32.20:

Таблица 32.20. Поля регистра IC_CLR_RX_OVER

Разряды	Название	Описание	Сброс
31:1	-	Резерв	0x0

Разряды	Название	Описание	Сброс
0	CLR_RX_OVER	Чтение этого регистра сбрасывает прерывание RX_OVER (бит 1 регистра IC_RAW_INTR_STAT)	0x0

32.2.20 Регистр IC_CLR_TX_OVER

Смещение:0x4c

Регистр сброса прерывания TX_OVER.

Формат регистра IC_CLR_TX_OVER приведен в Таблица 32.21:

Таблица 32.21. Поля регистра IC_CLR_TX_OVER

Разряды	Название	Описание	Сброс
31:1	-	Резерв	0x0
0	CLR_TX_OVER	Чтение этого регистра сбрасывает прерывание TX_OVER (бит 3 регистра IC_RAW_INTR_STAT)	0x0

32.2.21 Регистр IC_CLR_RD_REQ

Смещение:0x50

Регистр сброса прерывания RD_REQ.

Формат регистра IC_CLR_RD_REQ приведен в Таблица 32.22:

Таблица 32.22. Поля регистра IC_CLR_RD_REQ

Разряды	Название	Описание	Сброс
31:1	-	Резерв	0x0
0	CLR_RD_REQ	Чтение этого регистра сбрасывает прерывание RX_UNDER (бит 5 регистра IC_RAW_INTR_STAT)	0x0

32.2.22 Регистр IC_CLR_TX_ABRT

Смещение:0x54

Регистр сброса прерывания TX_ABRT.

Формат регистра IC_CLR_TX_ABRT приведен в Таблица 32.23:

Таблица 32.23. Поля регистра IC_CLR_TX_ABRT

Разряды	Название	Описание	Сброс
31:1	-	Резерв	0x0
0	CLR_TX_ABRT	Чтение этого регистра сбрасывает прерывание TX_ABRT (бит 6 регистра IC_RAW_INTR_STAT) и регистр IC_TX_ABRT_SOURCE. Reset value: 0x0	0x0

32.2.23 Регистр IC_CLR_RX_DONE

Смещение:0x58

Регистр сброса прерывания RX_DONE.

Формат регистра IC_CLR_RX_DONE приведен в Таблица 32.24:

Таблица 32.24. Поля регистра IC_CLR_RX_DONE

Разряды	Название	Описание	Сброс
31:1	-	Резерв	0x0
0	CLR_RX_DONE	Чтение этого регистра сбрасывает прерывание RX_DONE (бит 7 регистра IC_RAW_INTR_STAT)	0x0

32.2.24 Регистр IC_CLR_ACTIVITY

Смещение:0x5c

Регистр сброса прерывания ACTIVITY.

Формат регистра IC_CLR_ACTIVITY приведен в Таблица 32.25:

Таблица 32.25. Поля регистра IC_CLR_ACTIVITY

Разряды	Название	Описание	Сброс
31:1	-	Резерв	0x0
0	CLR_ACTIVITY	Чтение этого регистра сбрасывает прерывание ACTIVITY (бит 8 регистра IC_RAW_INTR_STAT), если контроллер I^2C не активен.	0x0

32.2.25 Регистр IC_CLR_STOP_DET

Смещение:0x60

Регистр сброса прерывания STOP_DET.

Формат регистра IC_CLR_STOP_DET приведен в Таблица 32.26:

Таблица 32.26. Поля регистра IC_CLR_STOP_DET

Разряды	Название	Описание	Сброс
31:1	-	Резерв	0x0
0	CLR_STOP_DET	Чтение этого регистра сбрасывает прерывание STOP_DET (бит 9 регистра IC_RAW_INTR_STAT)	0x0

32.2.26 Регистр IC_CLR_START_DET

Смещение:0x64

Регистр сброса прерывания START_DET.

Формат регистра IC_CLR_START_DET приведен в Таблица 32.27:

Таблица 32.27. Поля регистра IC_CLR_START_DET

Разряды	Название	Описание	Сброс
31:1	-	Резерв	0x0
0	CLR_START_DET	Чтение этого регистра сбрасывает прерывание START_DET (бит 10 регистра IC_RAW_INTR_STAT)	0x0

32.2.27 Регистр IC_CLR_GEN_CALL

Смещение:0x68

Регистр сброса прерывания GEN_CALL.

Формат регистра IC_CLR_GEN_CALL приведен в Таблица 32.28:

Таблица 32.28. Поля регистра IC_CLR_GEN_CALL

Разряды	Название	Описание	Сброс
31:1	-	Резерв	0x0
0	CLR_GEN_CALL	Чтение этого регистра сбрасывает прерывание GEN_CALL (11 бит регистра IC_RAW_INTR_STAT)	0x0

32.2.28 Регистр IC_ENABLE

Смещение:0x6c

Регистр включения I2C.

Формат регистра IC_ENABLE приведен в Таблица 32.29:

Таблица 32.29. Поля регистра IC_ENABLE

Разряды	Название	Описание	Сброс
31:19	-	Резерв	0x0
18	SMBUS_ALERT_EN	Бит SMBUS_ALERT_CTRL используется для управления установкой сигнала SMBALERT 1: Установка сигнала SMBALERT Этот бит сбрасывается автоматически после получения подтверждения от мастера по адресу Alert Response.	0x0
17	SMBUS_SUSPEND_EN	Бит используется для установки и снятия сигнала SMBSUS. 0: Сигнал SMBSUS снят. 1: Сигнал SMBSUS установлен.	0x0

Разряды	Название	Описание	Сброс
16	SMBUS_CLK_RESET	Бит используется в режиме SMBus Host для инициализации SMBus Master Clock Reset. Установка бита должна происходить только когда master находится в состояние idle.	0x0
15:4	-	Резерв	0x0
3	SDA_STUCK_RECOVERY_ENABLE	Бит управления восстановлением залипания на низком уровне линии SDA Если прерывание TX_ABORT указывает на залипание линии SDA на низком уровне (IC_TX_ABRT_SOURCE[17]), то установка этого бита в 1 приведет к старту SDA Recovery Mechanism (который отправит не более 9 тактов SCL и STOP-последовательность для освобождения линии SDA)	0x0
2	TX_CMD_BLOCK	Бит блокировки передатчика. В режиме mater: 1: Передача данных по шине I2C заблокирована независимо от наличия данных в TX FIFO. 0: Передача данных по шине I2C инициализируется, как только полный блок данных поступит в TX FIFO. Перед блокировкой передатчика необходимо убедиться, что TX FIFO пуст (IC_STATUS[2]==1) и Master в состоянии Idle (IC_STATUS[5]==0).	0x0
1	ABORT	При установке бита в 1, контроллер инициализирует прекращение транзакции. Бит позволяет программно прервать передачу данных в режиме master. Бит может быть установлен только при ENABLE==1. Однажды установленный в 1 бит ABORT не может быть программно сброшен в 0, он сбросится автоматически после прекращения транзакции.	/0x0
0	ENABLE	0: Контроллер выключен 1: Контроллер включен. При выключении контроллера, происходит следующее: • Стирается содержимое FIFO приемника и FIFO передатчика. • Статус-биты регистра IC_INTR_STAT остаются активными до тех пока контроллер не перейдет в состояние IDLE. Если контроллер функционировал как передатчик, он завершит текущую передачу и остановится из-за того, что буфер передатчика станет пуст. Если контроллер функционировал как приемник, то завершив текущую передачу, контроллер не подтвердит завершение передачи битом acknowledge.	0x0

32.2.29 Регистр IC_STATUS

Смещение:0x70

Регистр статуса контроллера

Регистр доступен только по чтению, отображает статус текущей передачи и статус FIFO. Регистр статуса может быть прочтен в любое время. Когда происходит выключение контроллера записью 0 в 0-ой бит регистра IC_ENABLE, происходит следующее:

- биты 1 и 2 устанавливаются в 1;
- биты 3 и 10 устанавливаются в 0;

Когда контроллер переходит в состояние IDLE:

- биты 5 и 6 устанавливаются в 0

Формат регистра IC_STATUS приведен в Таблица 32.30:

Таблица 32.30. Поля регистра IC_STATUS

Разряды	Название	Описание	Сброс
31:21	-	Резерв	0x0
20	SMBUS_ALERT_STATUS	Бит индикации статуса сигнала SMBus Suspend signal (is_smbalert_in_n).	0x0
19	SMBUS_SUSPEND_STATUS	Бит индикации статуса сигнала SMBus Suspend signal (is_smbsus_in_n).	0x0
18	SMBUS_SLAVE_ADDR_RESOLVED	Бит индикации разрешения Slave адреса ic_sar ARP Master.	0x0
17	SMBUS_SLAVE_ADDR_VALID	Бит индикации валидности slave адреса ic_sar.	0x0
16	SMBUS_QUICK_CMD_BIT	Установка этого бита указывает на получения команды Quick. Бит будет сброшен в 0 после чтения.	0x0
15:12	-	Резерв	0x0
11	SDA_STUCK_NOT_RECOVERED	Когда контроллер работает в режиме Master, установка этого бита в 1 указывает на то, что залипание линии SDA на низком уровне не было исправлено после работы механизма восстановления.	0x0
10	SLV_HOLD_RX_FIFO_FULL	Бит устанавливается в 1, когда контроллер в режиме Slave удерживает шину при полном RX FIFO в ожидании поступления данных по шине.	0x0
9	SLV_HOLD_TX_FIFO_EMPTY	Бит устанавливается в 1, когда контроллер в режиме Slave удерживает шину после по запроса на чтение при пустом Tx FIFO. Шина I2C удерживается до тех пор, пока в Tx FIFO не поступят данные для чтения.	0x0
8	MST_HOLD_RX_FIFO_FULL	Бит устанавливается в 1, когда контроллер в режиме Master удерживает шину при полном RX FIFO в ожидании поступления данных по шине.	0x0
7	MST_HOLD_TX_FIFO_EMPTY	Только для I2C0: Этот бит устанавливается в 1, когда контроллер в режиме Master удерживает шину на транзакции записи при пустом TX FIFO.	0x0
6	SLV_ACTIVITY	Slave FSM Activity Status. Статус активности slave состояния. 0: Состояние slave контроллера не активно	0x0

Разряды	Название	Описание	Сброс
		1: Состояние slave контроллера активно	
5	MST_ACTIVITY	Статус активности состояния master. 0: Состояние master контроллера не активно 1: Состояние master контроллера активно	0x0
4	RFF	Буфер приемника полон. 0 - Буфер приемника не полон 1 - Буфер приемника полон	0x0
3	RFNE	FIFO приемника не пуст. 0 - Буфер приемника пуст 1 - Буфер приемника не пуст	0x0
2	TFE	FIFO передатчика абсолютно пуст. 0 - Буфер передатчика не пуст 1 - Буфер передатчика пуст	0x1
1	TFNF	FIFO передатчика не полон. 0 - Буфер передатчика полон 1 - Буфер передатчика не полон	0x1
0	ACTIVITY	Статус активности шины I2C	0x0

32.2.30 Регистр IC_TXFLR

Смещение:0x74

Регистр уровня FIFO передатчика.

Этот регистр содержит число строк данных в FIFO передатчика. Регистр сбрасывается в следующих случаях:

- происходит выключение контроллера;
- происходит обрыв передачи. Устанавливается бит TX_ABRT регистра IC_RAW_INTR_STAT.

Формат регистра IC_TXFLR приведен в Таблица 32.31:

Таблица 32.31. Поля регистра IC_TXFLR

Разряды	Название	Описание	Сброс
31:7	Резерв	-	0x0
6:0	TXFLR	Уровень FIFO передатчика. Содержит число доступных записей данных в FIFO передатчика.	0x0

32.2.31 Регистр IC_RXFLR

Смещение:0x78

Регистр уровня FIFO приемника.

Этот регистр содержит число строк данных в FIFO приемника. Регистр сбрасывается в следующих случаях:

- происходит выключение контроллера;
- происходит обрыв передачи по любой причине, описанной в регистре IC_TX_ABRT_SOURCE.

Формат регистра IC_RXFLR приведен в Таблица 32.32:

Таблица 32.32. Поля регистра IC_RXFLR

Разряды	Название	Описание	Сброс
31:7	RSVD_RXFLR	-	0x0
6:0	RXFLR	Уровень FIFO приемника. Содержит число доступных записей данных в FIFO приемника.	0x0

32.2.32 Регистр IC_SDA_HOLD

Смещение:0x7c

Регистр времени удержания сигнала SDA.

Этот регистр контролирует время удержания (выраженное в числе тактов частоты i2c*x*_clk) сигнала SDA после заднего фронта сигнала SCL в обоих slave и master режимах работы контроллера. Значение этого регистра должно быть выше одного такта для режима master и 7-и тактов для режима slave. Регистр доступен по записи только при IC_ENABLED[0]=0.

Причем IC_SDA_HOLD ≤ N_SCL_LOW - 2, где N_SCL_LOW это продолжительность низкого уровня сигнала SCL в тактах i2c*x*_clk.

Формат регистра IC_SDA_HOLD приведен в Таблица 32.33:

Таблица 32.33. Поля регистра IC_SDA_HOLD

Разряды	Название	Описание	Сброс
31:24	-	Резерв	0x0
23:16	IC_SDA_RX_HOLD	Устанавливает требуемое значение времени удержания сигнала SDA в тактах i2c*x*_clk в режиме приемника.	0x0
15:0	IC_SDA_TX_HOLD	Устанавливает требуемое значение времени удержания сигнала SDA в тактах i2c*x*_clk в режиме передатчика.	0x1

32.2.33 Регистр IC_TX_ABRT_SOURCE

Смещение:0x80

Регистр причины обрыва передачи.

Этот регистр имеет 16 бит, которые отображают причину установки в «1» бита TX_ABRT. Регистр сбрасывается чтением регистра IC_CLR_TX_ABRT или IC_CLR_INTR (кроме бита

9 - ABRT_SBYTE_NORSTRT). Для сброса девятого бита, сначала должны быть выполнены следующие условия.

- должна быть включена возможность отправки последовательности RESTART, IC_CON[5]=1;
- должен быть сброшен бит SPECAL (IC_TAR[11]) или бит GC_OR_START (IC_TAR[10]).

После выполнения этих условий бит ABRT_SBYTE_NORSTRT сбрасывается также, как и остальные.

Формат регистра IC_TX_ABRT_SOURCE приведен в Таблица 32.34:

Таблица 32.34. Поля регистра IC_TX_ABRT_SOURCE

Разряды	Название	Описание	Источник	Сброс
31:23	TX_FLUSH_CNT	Поле хранит количество Tx FIFO Data Commands, которые были стерты из-за прерывания TX_ABRT.	Master-передатчик или Slave-передатчик	0x0
22:18	-	Резерв		0x0
17	ABRT_SDA_STUCK_AT_LOW	1: Master обнаружил залипание SDA на низком уровне на значение IC_SDA_STUCK_AT_LOW_TIMEOUT	Master	0x0
16	ABRT_USER_ABRT	1: Прерывание (abort) транзакции (IC_ENABLE[1])	Master-передатчик	0x0
15	ABRT_SLVRD_INTX	1: Процессор отвечает запросу от slave на передачу данных удаленному master -устройству, а пользователь пишет 1 в CMD (IC_DATA_CMD[8])	Slave - передатчик	0x0
14	ABRT_SLV_ARBLOST	1: Slave теряет шину во время передачи данных. В это же время устанавливается IC_TX_ABRT_SOURCE[12].	Slave - передатчик	0x0
13	ABRT_SLVFLUSH_TXFIFO	1: Slave получил команду на чтение и некоторые данные находятся в FIFO передатчика, поэтому slave запрашивает TX_ABRT прерывание для стирания старых данных из FIFO передатчика.	Slave-передатчик	0x0
12	ARB_LOST	1: Master проиграл арбитраж, или (если IC_TX_ABRT_SOURCE[14] также установлен) slave -передатчик проигрывает арбитраж. (Контроллер может находиться в режиме master и в режиме slave одновременно)	Master - передатчик или master -приемник	0x0
11	ABRT_MASTER_DIS	1: Пользователь пытается инициализировать master обмен при выключенном master -режиме.	Master - передатчик или master -приемник.	0x0
10	ABRT_10B_RD_NORSTRT	1: Отключена возможность отправлять последовательности RESTART (IC_RESTART_EN(IC_CON[5]) = 0) и master отправляет запрос на чтение в режиме 10-и битной адресации	Master - приемник	0x0
9	ABRT_SBYTE_NORSTRT	1: Отключена возможность отправлять последовательности RESTART	Master	0x0

Разряды	Название	Описание	Источник	Сброс
		(IC_RESTART_EN(IC_CON[5]) = 0) и пользователь пытается отправить START-байт.		
8	ABRT_HS_NORSTRT	1: Отключена возможность отправлять последовательности RESTART (IC_RESTART_EN(IC_CON[5]) = 0) и пользователь пытается в режиме high-speed master осуществить передачу данных.	Master - передатчик или master - приемник.	0x0
7	ABRT_SBYTE_ACKDET	1: Мастер отправил START-байт и получил сигнал подтверждения	Master	0x0
6	ABRT_HS_ACKDET	1: Контроллер в режиме master и скоростном режиме high-speed получил подтверждение на high-speed master code.	Master	0x0
5	ABRT_GCALL_READ	1: Контроллер в режиме мастера отправил General Call, но пользователь объявил запрос на чтение (IC_DATA_CMD[8]=1)	Master - передатчик	0x0
4	ABRT_GCALL_NOACK	1: Контроллер в режиме мастера отправил General Call, и ни один slave на шине не отправил подтверждающего сигнала.	Master - передатчик	0x0
3	ABRT_TXDATA_NOACK	1: Контроллер в режиме мастера получил сигнал подтверждения от slave после отправления адреса. Но после отправления байта(-ов) данных по этому адресу, он не получил сигнала подтверждения от slave -устройства.	Master - передатчик	0x0
2	ABRT_10ADDR2_NOACK	1: Контроллер в режиме master с 10-и битной адресацией не получил сигнала подтверждения от slave после отправления второго 10-и битного адреса	Master - передатчик или master - приемник.	0x0
1	ABRT_10ADDR1_NOACK	1: Контроллер в режиме master с 10-и битной адресацией не получил сигнала подтверждения от slave после отправления первого 10-и битного адреса	Master - передатчик или master - приемник.	0x0
0	ABRT_7B_ADDR_NOACK	1: Контроллер в режиме master с 7-и битной адресацией не получил сигнала подтверждения от slave после отправления адреса.	Master - передатчик или master - приемник.	0x0

32.2.34 Регистр IC_DMA_CR

Смещение:0x88

Регистр управления аппаратным интерфейсом запросов к PDMA.

Этот регистр используется для включения интерфейса взаимодействия с контроллером DMA. Регистр имеет отдельные биты на прием и на передачу. Запись в этот регистр возможна при включенном контроллере.

Формат регистра IC_DMA_CR приведен в Таблица 32.35:

Таблица 32.35. Поля регистра IC_DMA_CR

Разряды	Название	Описание	Сброс
31:2	-	Резерв	0x0
1	TDMAE	0: Выключение канала FIFO DMA передатчика. 1: Выключение канала FIFO DMA передатчика.	0x0
0	RDMAE	0: Выключение канала FIFO DMA приемника. 1: Выключение канала FIFO DMA приемника.	0x0

32.2.35 Регистр IC_DMA_TDLR

Смещение:0x8c

Регистр уровня данных передатчика DMA.

Формат регистра IC_DMA_TDLR приведен в Таблица 32.36:

Таблица 32.36. Поля регистра IC_DMA_TDLR

Разряды	Название	Описание	Сброс
31:6	-	Резерв	0x0
5:0	DMATDL	Контролирует уровень данных в буфере передатчика, по достижении которого будет возникать запрос к PDMA на запись данных в FIFO передатчика.	0x0

32.2.36 Регистр IC_DMA_RDLR

Смещение:0x90

Регистр уровня данных приемника DMA.

Формат регистра IC_DMA_RDLR приведен в Таблица 32.37:

Таблица 32.37. Поля регистра IC_DMA_RDLR

Разряды	Название	Описание	Сброс
31:6	-	Резерв	0x0
5:0	DMARDL	Контролирует уровень данных в буфере приемника, по достижении которого будет возникать запрос к PDMA на чтение данных из FIFO приемника.	0x0

32.2.37 Регистр IC_SDA_SETUP

Смещение:0x94

Регистр продолжительности SDA.

Этот регистр контролирует продолжительность временной задержки (в тактах частоты i2cx_clk) введенной между передним фронтом сигнала SCL относительно изменения SDA

Формат регистра IC_SDA_SETUP приведен в Таблица 32.38:

Таблица 32.38. Поля регистра IC_SDA_SETUP

Разряды	Название	Описание	Сброс
31:8	Резерв	-	0x0
7:0	SDA_SETUP	Настройка SDA	0x64

32.2.38 Регистр IC_ACK_GENERAL_CALL

Смещение:0x98

Регистр подтверждения General Call

Этот регистр контролирует, каким сигналом ACK или NACK, будет отвечать контроллер на General Call адрес.

Формат регистра IC_ACK_GENERAL_CALL приведен в Таблица 32.39:

Таблица 32.39. Поля регистра IC_ACK_GENERAL_CALL

Разряды	Название	Описание	Сброс
31:1	-	Резерв	0x0
0	ACK_GEN_CALL	0: NACK 1: ACK	0x1

32.2.39 Регистр IC_ENABLE_STATUS

Смещение:0x9c

Этот регистр используется для получения информации о статусе контроллера, когда регистр IC_ENABLE переключается из 1 в 0, т.е. когда происходит выключение контроллера.

Если IC_ENABLE установлен в 1, то биты 1:2 преобразуются в 0, а бит 0 в 1.

Если IC_ENABLE установлен в 0, то биты 1:2 будут действительны тогда, когда бит 0 будет читаться как 0.

Формат регистра IC_ENABLE_STATUS приведен в Таблица 32.40:

Таблица 32.40. Поля регистра IC_ENABLE_STATUS

Разряды	Название	Описание	Сброс
31:3	-	Резерв	0x0
2	SLV_RX_DATA_LOST	Slave потерял принимаемые данные. Этот бит устанавливается, если прерывается операция slave-приемника с потерей байта данных из-за выключения контроллера (изменения состояния регистра IC_ENABLE из 1 в 0) в момент приема байта данных. Когда это	0x0

Разряды	Название	Описание	Сброс
		происходит, контроллер выдает сигнал NACK по завершению приема. CPU может безопасно читать этот бит, когда IC_EN (бит 0) читается как 0.	
1	SLV_DISABLED_WHILE_BUSY	Slave выключен когда занят (передача, прием) Этот бит отображает обрыв текущей операции контроллера, работающего в режиме slave из-за выключения контроллера (изменение состояния регистра IC_ENABLE из 1 в 0). Этот бит устанавливается, если CPU пишет 0 в регистр IC_ENABLE, когда: • Контроллер принимал байт адреса как slave-передатчик от удаленного master-устройства. • Конотроллер принимал байты адреса и данных как slave-приемник от удаленного master -устройства. Контроллер выдает сигнал NACK по завершению текущей фазы обмена в любом случае. CPU может безопасно читать этот бит, когда IC_EN (бит 0) читается как 0.	0x0
0	IC_EN	0: Контроллер выключен 1: Контроллер включен CPU может безопасно читать этот бит в любое время. Когда этот бит читается как 0, CPU может безопасно читать SLV_RX_DATA_LOST (бит 2) и SLV_DISABLED_WHILE_BUSY (бит 1).	0x0

32.2.40 Регистр IC_HS_SPKLEN

Смещение:0xa4

Регистр настройки параметров фильтрации помех для high-speed режима.

Этот регистр используется для хранения продолжительности (измеренной в кол-ве тактов ic_clk) наиболее длинной фильтруемой помехи.

Формат регистра IC_HS_SPKLEN приведен в Таблица 32.41:

Таблица 32.41. Поля регистра IC_HS_SPKLEN

Разряды	Название	Описание	Сброс
31:8	-	Резерв	0x0
7:0	IC_HS_SPKLEN	Регистр доступен для записи только при выключенном контроллере (IC_ENABLE=0). Регистр устанавливает продолжительность наиболее длинной фильтруемой помехи в линиях SDA и SCL (измеренной в кол-ве тактов i2cx_clk).	0x2

32.2.41 Регистр IC_FS_SPKLEN

Смещение:0xa0

Регистр настройки параметров фильтрации помех для standard-speed и fast-speed режимов.

Этот регистр используется для хранения продолжительности (измеренной в кол-ве тактов ic_clk) наиболее длинной фильтруемой помехи.

Формат регистра IC_FS_SPKLEN приведен в Таблица 32.42:

Таблица 32.42. Поля регистра IC_FS_SPKLEN

Разряды	Название	Описание	Сброс
31:8	-	Резерв	0x0
7:0	IC_FS_SPKLEN	Регистр доступен для записи только при выключенном контроллере (IC_ENABLE=0). Регистр устанавливает продолжительность наиболее длинной фильтруемой помехи в линиях SDA и SCL (измеренной в кол-ве тактов i2cx_clk).	0xa

32.2.42 Регистр IC_CLR_RESTART_DET

Смещение:0xa8

Регистр сброса прерывания RESTART_DET.

Формат регистра IC_CLR_RESTART_DET приведен в Таблица 32.43:

Таблица 32.43. Поля регистра IC_CLR_RESTART_DET

Разряды	Название	Описание	Сброс
31:1	-	Резерв	0x0
0	CLR_RESTART_DET	Чтение этого регистра сбрасывает прерывание RESTART_DET (бит 12 регистра RAW_INTR_STAT)	0x0

32.2.43 Регистр IC_SCL_STUCK_AT_LOW_TIMEOUT

Смещение:0xac

I2C SCL Stuck at Low Timeout

Только для I2C0.

Регистр используется для хранения продолжительности залипания линии SCL на низком уровне, измеренной в тактах частоты ic_clk, после которого происходит установка прерывания SCL_STUCK_AT_LOW.

Формат регистра IC_SCL_STUCK_AT_LOW_TIMEOUT приведен в Таблица 32.44:

Таблица 32.44. Поля регистра IC_SCL_STUCK_AT_LOW_TIMEOUT

Разряды	Название	Описание	Сброс
31:0	IC_SCL_STUCK_LOW_TIMEOUT	Регистр принимает значение таймаута по залипанию линии SCL на низком уровне для установки прерывания SCL_STUCK_AT_LOW. Регистр может быть записан только при выключенном контроллере (IC_ENABLE[0]=0)	0xffffffff

32.2.44 Регистр IC_SDA_STUCK_AT_LOW_TIMEOUT

Смещение:0xb0

Регистр хранит значение таймаута, выраженного в тактах частоты ic_clk. В случае, если линия SDA находится это количество тактов в низком уровне, может произойти восстановление линии через отправку импульса SCL.

Формат регистра IC_SDA_STUCK_AT_LOW_TIMEOUT приведен в Таблица 32.45:

Таблица 32.45. Поля регистра IC_SDA_STUCK_AT_LOW_TIMEOUT

Разряды	Название	Описание	Сброс
31:0	IC_SDA_STUCK_LOW_TIMEOUT	Контроллер инициализирует восстановление линии SDA, в зависимости от значения бита SDA_STUCK_RECOVERY_EN (IC_ENABLE[3]), если детектировано залипание линии SDA на указанное количество тактов частоты ic_clk.	0xffffffff

32.2.45 Регистр IC_CLR_SCL_STUCK_DET

Смещение:0xb4

Регистр сброса прерывания по залипанию линии SCL на низком уровне.

Формат регистра IC_CLR_SCL_STUCK_DET приведен в Таблица 32.46:

Таблица 32.46. Поля регистра IC_CLR_SCL_STUCK_DET

Разряды	Название	Описание	Сброс
31:1	-	Резерв	0x0
0	CLR_SCL_STUCK_DET	Чтение этого регистра сбрасывает прерывание SCL_CTUCT_AT_LOW (бит 15 регистра IC_RAW_INTR_STAT).	0x0

32.2.46 Регистр IC_SMBUS_CLK_LOW_SEXT

Смещение:0xbc

Только для I2C0.

Этот регистр содержит значение расширения таймаута, используемое для определения Slave Clock Extend таймаута для одной передачи (от START до STOP). Этот регистр может быть записан, только когда контроллер выключен (IC_ENABLE[0]=0).

Формат регистра IC_SMBUS_CLK_LOW_SEXT приведен в Таблица 32.47:

Таблица 32.47. Поля регистра IC_SMBUS_CLK_LOW_SEXT

Разряды	Название	Описание	Сброс
31:0	SMBUS_CLK_LOW_SEXT_TIMEOUT	Поле принимает значение tLOW-SEXT в режиме master, выраженное в количестве тактов ic_clk	0xffffffff

32.2.47 Регистр IC_SMBUS_CLK_LOW_MEXT

Смещение:0xc0

SMBus Master Clock Extend Timeout Register

Только для I2C0.

Этот регистр содержит значение расширения таймаута, используемое для определения Master Clock Extend таймаута для одной передачи (от START до STOP). Этот регистр может быть записан, только когда контроллер выключен (IC_ENABLE[0]=0).

Формат регистра IC_SMBUS_CLK_LOW_MEXT приведен в Таблица 32.48:

Таблица 32.48. Поля регистра IC_SMBUS_CLK_LOW_MEXT

Разряды	Название	Описание	Сброс
31:0	SMBUS_CLK_LOW_MEXT_TIMEOUT	Поле принимает значение tLOW-MEXT в режиме master, выраженное в количестве тактов ic_clk	0xffffffff

32.2.48 Регистр IC_SMBUS_THIGH_MAX_IDLE_COUNT

Смещение:0xc4

Регистр максимального значения для счетчика THigh Bus-idle. Регистр устанавливает значение счетчика простоя шины Bus-idle, который используется динамического добавления мастера к шине или в случае, когда мастер генерирует сброс частоты на шине. Регистр используется для хранения продолжительности простоя шины, измеренной в тактах ic_clk, для определения состояния Bus Idle когда линии SCL и SDA остаются в высоком состоянии указанный интервал времени.

Формат регистра IC_SMBUS_THIGH_MAX_IDLE_COUNT приведен в Таблица 32.49:

Таблица 32.49. Поля регистра IC_SMBUS_THIGH_MAX_IDLE_COUNT

Разряды	Название	Описание	Сброс
31:16	RSVD_SMBUS_THIGH_MAX_BUS_IDLE_CNT	SMBUS_THIGH_MAX_BUS_IDLE_CNT Reserved bits - Read Only	0x0
15:0	SMBUS_THIGH_MAX_BUS_IDLE_CNT	Поле используется для установки периода Master Thigh MAX Bus-Idle.	0xffff

32.2.49 Регистр IC_SMBUS_INTR_STAT

Смещение:0xc8

Регистр статуса прерывания SMBUS.

Только для I2C0

Каждый бит этого регистра имеет соответствующий маскирующий бит в регистре IC_SMBUS_INTR_MASK. Значения битов прерывания до маскирования доступны в регистре IC_SMBUS_RAW_INTR_STAT.

Формат регистра IC_SMBUS_INTR_STAT приведен в Таблица 32.50:

Таблица 32.50. Поля регистра IC_SMBUS_INTR_STAT

Разряды	Название	Описание	Сброс
31:11	-	Резерв	0x0
10	R_SMBUS_ALERT_DET	См. регистр IC_SMBUS_RAW_INTR_STAT	0x0
9	R_SMBUS_SUSPEND_DET		0x0
8	R_SLV_RX_PEC_NACK		0x0
7	R_ARP_ASSGN_ADDR_CMD_DET		0x0
6	R_ARP_GET_UDID_CMD_DET		0x0
5	R_ARP_RST_CMD_DET		0x0
4	R_ARP_PREPARE_CMD_DET		0x0
3	R_HOST_NOTIFY_MST_DET		0x0
2	R_QUICK_CMD_DET		0x0
1	R_MST_CLOCK_EXTND_TIMEOUT		0x0
0	R_SLV_CLOCK_EXTND_TIMEOUT		0x0

32.2.50 Регистр IC_SMBUS_INTR_MASK

Смещение:0xcc

Регистр маски прерывания SMBUS.

Формат регистра IC_SMBUS_INTR_MASK приведен в Таблица 32.51:

Таблица 32.51. Поля регистра IC_SMBUS_INTR_MASK

Разряды	Название	Описание	Сброс
31:11	-	Резерв	0x0
10	M_SMBUS_ALERT_DET	См. регистр IC_SMBUS_RAW_INTR_STAT	0x1
9	M_SMBUS_SUSPEND_DET		0x1
8	M_SLV_RX_PEC_NACK		0x1
7	M_ARP_ASSGN_ADDR_CMD_DET		0x1
6	M_ARP_GET_UDID_CMD_DET		0x1
5	M_ARP_RST_CMD_DET		0x1
4	M_ARP_PREPARE_CMD_DET		0x1
3	M_HOST_NOTIFY_MST_DET		0x1
2	M_QUICK_CMD_DET		0x1
1	M_MST_CLOCK_EXTND_TIMEOUT		0x1
0	M_SLV_CLOCK_EXTND_TIMEOUT		0x1

32.2.51 Регистр IC_SMBUS_RAW_INTR_STAT

Смещение:0xd0

Регистр немаскированного статуса прерывания SMBus. Формат регистра IC_SMBUS_RAW_INTR_STAT приведен в Таблица 32.52:

Таблица 32.52. Поля регистра IC_SMBUS_RAW_INTR_STAT

Разряды	Название	Описание	Сброс
31:11	-	Резерв	0x0
10	SMBUS_ALERT_DET	Бит указывает на то, что Slave установил в 0 сигнал ic_smbalert_in_n.	0x0
9	SMBUS_SUSPEND_DET	Бит указывает на то, что Host установил в 0 сигнал ic_smbus_in_n.	0x0
8	SLV_RX_PEC_NACK	Бит указывает на то, что во время работы ARP slave произошла ошибка PEC и была отправлена NACK команда.	0x0
7	ARP_ASSGN_ADDR_CMD_DET	Бит указывает на получение команды Assign Address ARP.	0x0
6	ARP_GET_UDID_CMD_DET	Бит указывает на получение Get UDID ARP команды.	0x0
5	ARP_RST_CMD_DET	Бит указывает на получение команд General или Directed Reset ARP.	0x0
4	ARP_PREPARE_CMD_DET	Бит указывает на получение команды prepare to ARP.	0x0
3	HOST_NTFY_MST_DET	Бит указывает на получение команды Notify ARP Master ARP команды.	0x0
2	QUICK_CMD_DET	Бит указывает на получение Quick-команды по SMBus интерфейсов независимо от того, работает ли контроллер в режиме Master или Slave.	0x0
1	MST_CLOCK_EXTND_TIMEOUT	Бит указывает на расширение транзакции Master’ом (START-to-ACK, ACK-to-ACK, ACK-to-STOP) от START до STOP на IC_SMBUS_CLOCK_LOW_MEXT тактов ic_clk для каждого байта передачи.	0x0
0	SLV_CLOCK_EXTND_TIMEOUT	Бит указывает на расширение транзакции от Slave’а (от START до STOP) на IC_SMBUS_CLK_LOW_SEXT тактов ic_clk.	0x0

32.2.52 Регистр IC_CLR_SMBUS_INTR

Смещение:0xd4

Регистр сброса прерывания SMBus.

Формат регистра IC_CLR_SMBUS_INTR приведен в Таблица 32.53:

Таблица 32.53. Поля регистра IC_CLR_SMBUS_INTR

Разряды	Название	Описание	Сброс
31:11	-	Резерв	0x0
10	CLR_SMBUS_ALERT_DET	См. регистр IC_SMBUS_RAW_INTR_STAT	0x0
9	CLR_SMBUS_SUSPEND_DET		0x0
8	CLR_SLV_RX_PEC_NACK		0x0
7	CLR_ARP_ASSGN_ADDR_CMD_DET		0x0

Разряды	Название	Описание	Сброс
6	CLR_ARP_GET_UDID_CMD_DET		0x0
5	CLR_ARP_RST_CMD_DET		0x0
4	CLR_ARP_PREPARE_CMD_DET		0x0
3	CLR_HOST_NOTIFY_MST_DET		0x0
2	CLR_QUICK_CMD_DET		0x0
1	CLR_MST_CLOCK_EXTND_TIMEOUT		0x0
0	CLR_SLV_CLOCK_EXTND_TIMEOUT		0x0

32.2.53 Регистр IC_OPTIONAL_SAR

Смещение:0xd8

Дополнительный регистры Slave-адреса.

Только для I2C0.

Дополнительный регистр Slave-адреса для I2C в режиме SMBus.

Формат регистра IC_OPTIONAL_SAR приведен в Таблица 32.54:

Таблица 32.54. Поля регистра IC_OPTIONAL_SAR

Разряды	Название	Описание	Сброс
31:7	-	Резерв	0x0
6:0	OPTIONAL_SAR	Дополнительный регистр Slave-адреса для I2C в режиме SMBus	0x0

32.2.54 Регистр IC_SMBUS_UDID_LSB

Смещение:0xdc

Регистр SMBUS ARP UDID LSB

Регистр используется для хранения LSB 32-х битного значения UIDID Slave, используемого в SBus Address Resolution протоколе. Значение может быть записано только при выключенном контроллере (IC_ENABLE[0]=0)

Формат регистра IC_SMBUS_UDID_LSB приведен в Таблица 32.55:

Таблица 32.55. Поля регистра IC_SMBUS_UDID_LSB

Разряды	Название	Описание	Сброс
31:0	SMBUS_UDID_LSB	Регистр используется для хранения LSB 32-х битного значения UIDID Slave, используемого в SBus Address Resolution протоколе.	0xffffffff

32.2.55 Регистр IC_COMP_PARAM_1

Смещение:0xf4

Component Parameter Register 1

Формат регистра IC_COMP_PARAM_1 приведен в Таблица 32.56:

Таблица 32.56. Поля регистра IC_COMP_PARAM_1

Разряды	Название	Описание	Сброс
31:24	RSVD_IC_COMP_PARAM_1	IC_COMP_PARAM_1 Reserved bits - Read Only	0x0
23:16	TX_BUFFER_DEPTH	The value of this register is derived from the IC_TX_BUFFER_DEPTH coreConsultant parameter. - 0x00 = Reserved - 0x01 = 2 - 0x02 = 3 - ... - 0xFF = 256	0x3f
15:8	RX_BUFFER_DEPTH	The value of this register is derived from the IC_RX_BUFFER_DEPTH coreConsultant parameter. - 0x00: Reserved - 0x01: 2 - 0x02: 3 - ... - 0xFF: 256	0x3f
7	ADD_ENCODED_PARAMS	The value of this register is derived from the IC_ADD_ENCODED_PARAMS coreConsultant parameter. Reading 1 in this bit means that the capability of reading these encoded parameters via software has been included. Otherwise, the entire register is 0 regardless of the setting of any other parameters that are encoded in the bits.	0x1
6	HAS_DMA	The value of this register is derived from the IC_HAS_DMA coreConsultant parameter.	0x1
5	INTR_IO	The value of this register is derived from the IC_INTR_IO coreConsultant parameter.	0x1
4	HC_COUNT_VALUES	The value of this register is derived from the IC_HC_COUNT VALUES coreConsultant parameter.	0x0
3:2	MAX_SPEED_MODE	The value of this register is derived from the IC_MAX_SPEED_MODE coreConsultant parameter. - 0x0: Reserved - 0x1: Standard - 0x2: Fast - 0x3: High	0x2
1:0	APB_DATA_WIDTH	The value of this register is derived from the APB_DATA_WIDTH coreConsultant parameter.	0x2

32.2.56 Регистр IC_COMP_VERSION

Смещение:0xf8

I²C Component Version Register

Формат регистра IC_COMP_VERSION приведен в Таблица 32.57:

Таблица 32.57. Поля регистра IC_COMP_VERSION

Разряды	Название	Описание	Сброс
31:0	IC_COMP_VERSION	Specific values for this register are described in the Releases Table in the DW_apb_i2c Release Notes	0x3230312a

32.2.57 Регистр IC_COMP_TYPE

Смещение:0xfc

I2C Component Type Register

Формат регистра IC_COMP_TYPE приведен в Таблица 32.58:

Таблица 32.58. Поля регистра IC_COMP_TYPE

Разряды	Название	Описание	Сброс
31:0	IC_COMP_TYPE	Designware Component Type number = 0x44_57_01_40. This assigned unique hex value is constant and is derived from the two ASCII letters 'DW' followed by a 16-bit unsigned number.	0x44570140

32.3 Функционирование

32.3.1 Поведение I²C

I²C может управляться программно, чтобы быть либо:

- I²C мастер, взаимодействующий с I²C слейвами;
- I²C слейв, взаимодействующий с еще одним I²C мастером.

Мастер отвечает за генерацию тактовых импульсов и управление передачей данных. Слейв отвечает за передачу или прием данных от/к мастеру. Подтверждение данных отправляется устройством, принимающим данные, которое может быт как мастером, так и слейвом.

У каждого слейва есть свой уникальный адрес, который определяется разработчиком системы. Когда мастер хочет установить связь со слейвом, мастер передает START/RESTART, за которым следует адрес слейва и управляющий бит (R/W), чтобы

определить, хочет ли мастер передавать данные или получат данные от слейва. После адреса, слейв посылает ACK.

Если мастер пишет слейву, то принимающее устройство получает один байт данных. Эта транзакция продолжается до тех пор, пока мастер не завершит передачу с условием STOP. Если мастер считывает данные со слейва, то слейв передает байт данных мастеру, а затем мастер подтверждает транзакцию с помощью ACK. Эта транзакция продолжается до тех пор, пока мастер не завершит передачу (NACK), после получения последнего байта, а затем мастер выдает условие STOP или обращается к другому слейву после выдачи условия RESTART (Рисунок 32.1).

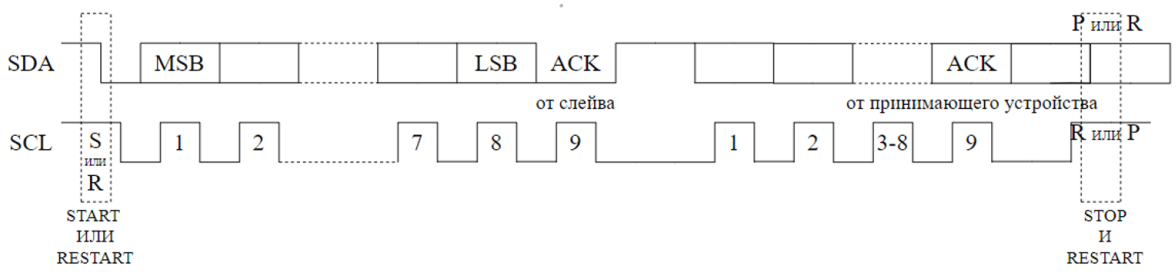


Рисунок 32.1. Передача данных по шине I²C

В сверхбыстром режиме мастер может выполнять только передачу записи на слейв устройства с NACK от слейв устройств. Передача данных на чтение в этом режиме запрещена.

I²C — это синхронный последовательный интерфейс. SDA — двунаправленный сигнал и изменяется только при низком уровне SCL, за исключением STOP, START и RESTART. Максимальное количество устройств на шине ограничено только емкостью 400 пФ. Данные передаются байтовыми пакетами.

32.3.2 Start и Stop генерация

При работе I²C мастера, ввод данных в передающее FIFO приводит к тому, что UART генерирует условие START на шине I²C. Если IC_EMPTYFIFO_HOLD_MASTER_EN установлен в 0, то опустошение передающего FIFO вызывает генерацию STOP на шине I²C. Если IC_EMPTYFIFO_HOLD_MASTER_EN и IC_DATA_CMD[9] установлены в 1, то генерируется состояние STOP на шине I²C. STOP не генерируется, если бит IC_DATA_CMD[9] равняется 0, даже если передающее FIFO пустое.

При работе слейва, I²C не генерирует условия START и STOP в соответствии с протоколом. Однако, если запрос чтения сделан к I²C, он удерживает SCL на низком уровне до тех пор, пока ему не будут переданы данные чтения. Это останавливает шины до тех пор, пока

данные считывания не будут предоставлены слейву I²C, или I²C слейв не будет отключено путем записи 0 в IC_ENABLE[0].

32.3.3 Комбинированные форматы

I²C поддерживает смешанные транзакции чтения и записи комбинированного формата как в 7-битном, так и в 10-битном режимах адресации.

I²C не поддерживает смешанные адреса и смешанный формат адресов, то есть транзакцию с 7-битовым адресом, за которой следует транзакция с 10-битным адресом, или наоборот – транзакции с комбинированным форматом.

Чтобы инициировать передачу комбинированного формата (IC_CON), IC_RESTART_EN должен быть равен 1. Если IC_RESTART_EN установлен и работает, как мастер, то в процессе завершения передачи, I²C проверяет передающее FIFO и выполняет следующую передачу. Если направления данного перевода отличается от направления предыдущего, то для выдачи перевода используется комбинированный формат. Если передающее FIFO пустое, когда текущая передача I²C завершена – в зависимости от значения IC_EMPTYFIFO_HOLD_MASTER_EN:

- Состояние STOP;
- IC_DATA_CMD[9]:
    - 1 — бит STOP.
    - 0 — SCL удерживается на низком уровне до тех пор, пока следующая команда не будет записана в передающее FIFO.

32.4 Протокол I²C

32.4.1 Последовательности Start и Stop

Когда шина I²C неактивна (находится в состоянии IDLE), оба сигнала SCL и SDA идут высоким уровнем. Когда мастер инициализирует передачу, контроллер отправляет последовательность Start: изменение уровня сигнала SDA с 1 на 0, при высоком уровне сигнала SCL. Когда мастер завершает передачу, контроллер отправляет последовательность Stop: изменение уровня сигнала SDA с 0 на 1, при высоком уровне сигнала SCL. Во всех остальных случаях значение сигнала SDA остается неизменным при высоком уровне сигнала SCL. Последовательности Start и Stop представлены на Рисунок 32.2.

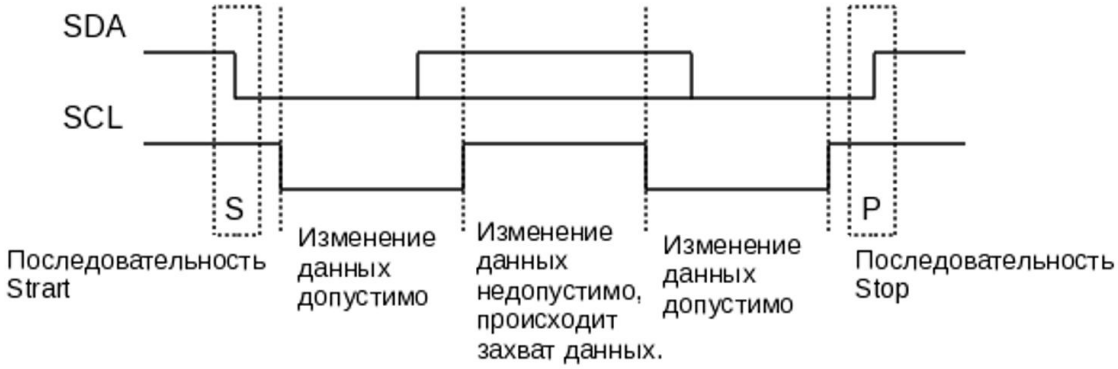


Рисунок 32.2. Последовательности Start и Stop

32.4.2 Протокол адресации слейва

Существует два формата адреса:

- 7 – битный;
- 10 – битный.

32.4.3 7-битный формат адреса

В 7 – битном формате адреса, первые семь бит первого байта устанавливают адрес слейва, а LSB является битом R/W, как показано на Рисунок 32.3. Когда LSB равняется 0, то мастер пишет в слейв. Когда LSB равняется 1, то мастер читает со слейва.

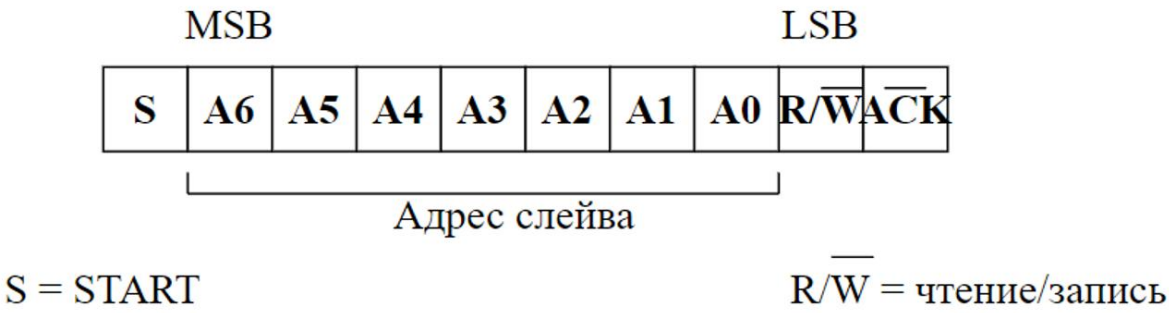


Рисунок 32.3. 7-битный формат адреса

32.4.4 10-битный формат адреса

При 10 – битной адресации передаются два байта для установки 10 – битного адреса. Первые пять бит сообщают слейву, что это 10 – битная передача, следующие 2 биты

устанавливают биты [9:8] слейв адреса. LSB является битом чтения/записи. Первые семь бит второго байта устанавливают адрес слейва. На Рисунок 32.4 показана 10 – битная передача.

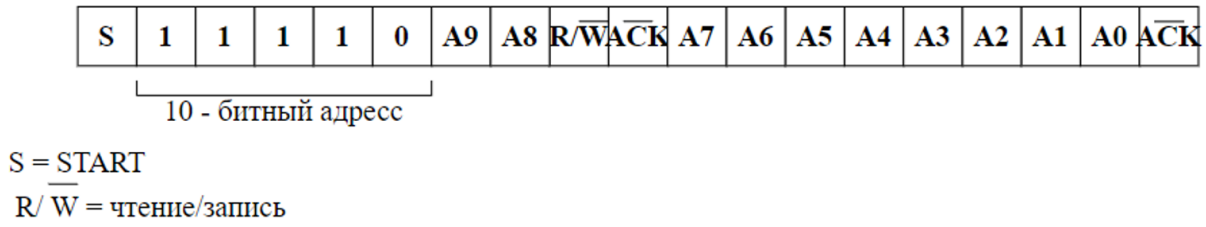


Рисунок 32.4. 10-битный формат адреса

В Таблица 32.59 определены специальные и зарезервированные адреса первого байта. I²C не ограничивает в использовании зарезервированных адресов. Однако, если использовать данные адреса, может обнаружиться несовместимость с другими компонентами I²C.

Таблица 32.59. I²C/SMBus значение бит в первом байте

Адрес слейва	Бит R/W	Описание
0000 000	0	Общий адрес (General Call Address). I²C помещает данные в принимающий буфер и выдает прерывание (General Call interrupt)
0000 000	1	START бит
0000 001	X	Адрес CBUS
0000 010	X	Резерв
0000 011	X	Резерв
0000 1XX	X	High-speed master code
1111 1XX	X	Резерв
1111 0XX	X	10 – битный слейв адрес
0001 000	X	SMbus Host
0001 100	X	SMBus Alert Response Address
1100 001	X	SMBus Device Default Address

32.4.5 Протокол передачи и приема данных



Рисунок 32.5. Обмен данными между мастер-передатчиком и слейв-приемником

Мастер-передатчик и слейв-приемник. Данные передаются побайтово, число байтов в одной передаче не ограничено. После того, как мастер-устройство отправляет слейв-устройству адрес с битом R/W или байт данных, слейв-приемник должен выдать импульс подтверждения ACK в линию SDA. Если слейв-приемник не отвечает импульсом ACK, мастер-устройство прерывает передачу и отправляет последовательность Stop. Структура обмена данными между мастер-передатчиком и слейв-приемником представлена на Рисунок 32.5.

Мастер-приемник и слейв-передатчик. Мастер принимает данные, как показано на Рисунок 32.6, тогда он подтверждает принятые от слейва данные импульсом ACK и ожидает следующий байт. В случае, если мастер желает прекратить передачу, то по окончании приема текущего байта он выдает сигнал NACK в линию SDA а затем отправляет последовательность Stop. Структура обмена между мастер-приемником и слейв-передатчиком представлена на Рисунок 32.6.

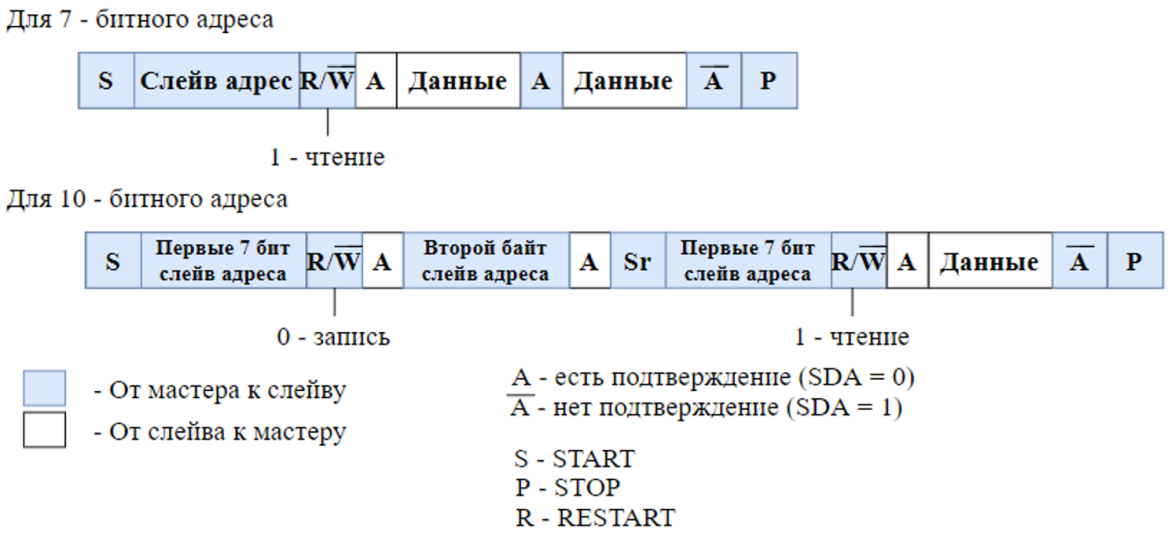


Рисунок 32.6. Обмен данными между мастер-передатчиком и слейв-приемником

Отправляя последовательность Restart вместо последовательности Stop, контроллер, работающий в режиме мастер, меняет направление передачи данных без утраты контроля шины.

32.4.6 Протокол START BYTE

Протокол START BYTE нужен для взаимодействия контроллера I^2C с устройствами, не имеющими аппаратного модуля I^2C. Использование START BYTE протокола имеет смысл только тогда, когда контроллер работает в режиме мастер.

Протокол заключается в следующем: для инициализации обмена контроллер, отправив последовательность Start, отправляет START-байт «00000001». Это необходимо для того, чтобы устройство, не имеющее аппаратного модуля I^2C, могло опрашивать шину с более низкой частотой до момента обнаружения одного из нулей START BYTE. Структура протокола START BYTE представлена на Рисунок 32.7.

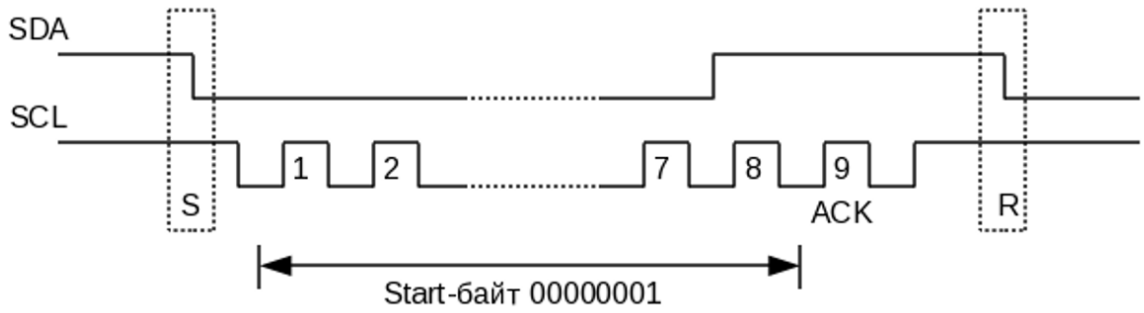


Рисунок 32.7. START BYTE

Полная последовательность действий контроллера:

1. Отправляет последовательность START.
2. Отправляет Start-байт («00000001»).
3. Отправляет импульс ACK.
4. Ни одно slave-устройство не отвечает импульсом ACK.
5. Отправляет последовательность RESTART.

Аппаратный приемник не отвечает импульсом ACK, поскольку Start-байт является зарезервированным адресом.

32.5 Управление Tx FIFO и генерация START, STOP и RESTART

При работе в качестве мастера компонент I²C поддерживает два режима управления Tx FIFO. Для выбора определенного режима используется параметр IC_EMPTYFIFO_HOLD_MASTER_EN.

32.5.1 Управление Tx FIFO, когда IC_EMPTYFIFO_HOLD_MASTER_EN = 0

Состояние STOP генерируется, если Tx FIFO становится пустым. Если включена возможность генерации RESTART, то генерируется RESTART, когда направления в Tx FIFO изменяется с чтения на запись или наоборот. Если RESTART не включен, то сначала генерируется STOP, за которым следует START.

На Рисунок 32.8 показаны биты регистра IC_DATA_CMD:



DATA - Поле Read/Write;

данные, полученные от слейва, считываются из этого поля;
в это поле записываются данные для отправки на слейв.

CMD - поле только для записи;

этот бит определяет, какая передача будет осуществляться - чтение (CMD=1) или запись (CMD=0).

Рисунок 32.8. Регистр IC_DATA_CMD

На Рисунок 32.9 показана временная диаграмма, иллюстрирующая поведение I²C, когда Tx FIFO становится пустым при работе в качестве мастер-передатчика.

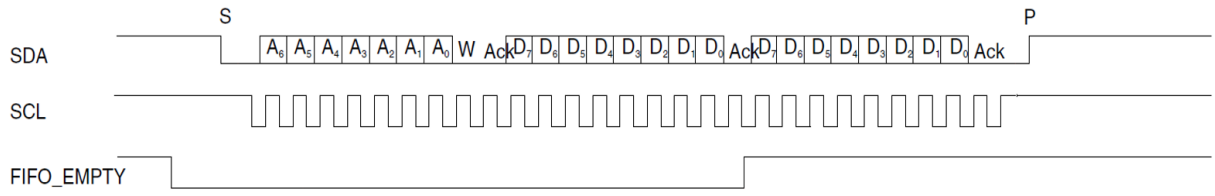


Рисунок 32.9. Поведение I²C, когда Tx FIFO становится пустым при работе в качестве мастер-передатчика

На Рисунок 32.10 показана временная диаграмма, иллюстрирующая поведение I²C, когда Tx FIFO становится пустым при работе в качестве мастер-приемника.

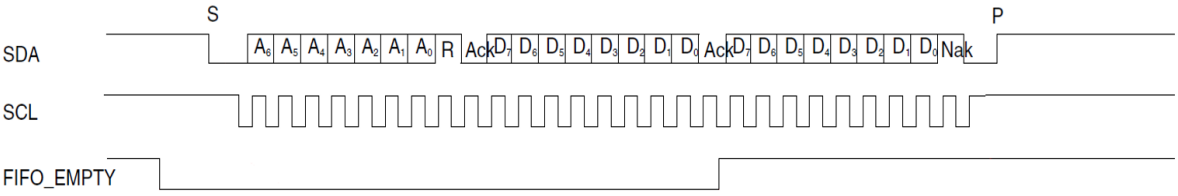


Рисунок 32.10. Поведение I²C, когда Tx FIFO становится пустым при работе в качестве мастер-приемника

32.5.2 Управление Tx FIFO, когда IC_EMPTYFIFO_HOLD_MASTER_EN = 1

STOP не генерируется, если Tx FIFO пустой, в этой ситуации удерживается SCL на низком уровне, останавливая шину до тех пор, пока в Tx FIFO не появится новая запись. STOP генерируется, когда пользователь специально запрашивает его.

На Рисунок 32.11 показаны биты регистра IC_DATA_CMD:

IC_DATA_CMD	RESTART	STOP	CMD	DATA
	10	9	8	7 0

DATA - Поле Read/Write;

данные, полученные от слейва, считываются из этого поля;
в это поле записываются данные для отправки на слейв.

CMD - поле только для записи;

бит определяет, какая передача будет осуществляться - чтение (CMD=1) или запись (CMD=0).

STOP - поле только для записи;

бит определяет, генерируется ли STOP после отправки или получения байта данных

RESTART - поле только для записи;

бит определяет, будет ли перезапуск (или
STOP с последующим START в случае, если возможность перезапуска не включена) генерируется перед отправкой или получением байта данных

Рисунок 32.11. Регистр IC_DATA_CMD

На Рисунок 32.12 показана временная диаграмма, иллюстрирующая поведение I^{2}C, когда Tx FIFO становится пустым при работе в качестве мастер-передатчика, а также показывает генерацию условия STOP.

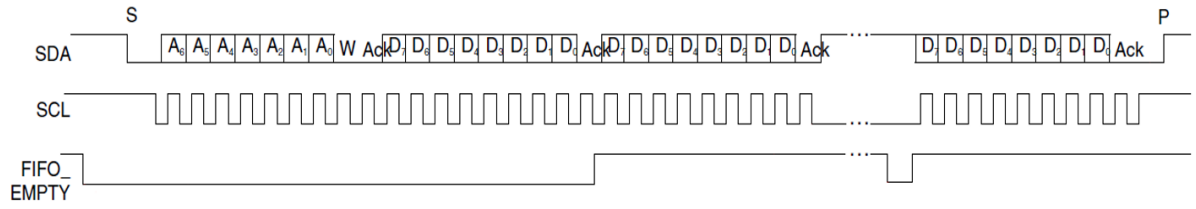


Рисунок 32.12. Поведение I^{2}C, когда Tx FIFO становится пустым при работе в качестве мастер-передатчика, генерация STOP

На Рисунок 32.13 показана временная диаграмма, иллюстрирующая поведение I^{2}C, когда Tx FIFO становится пустым при работе в качестве мастер-приемника, а также показывает генерацию условия STOP.

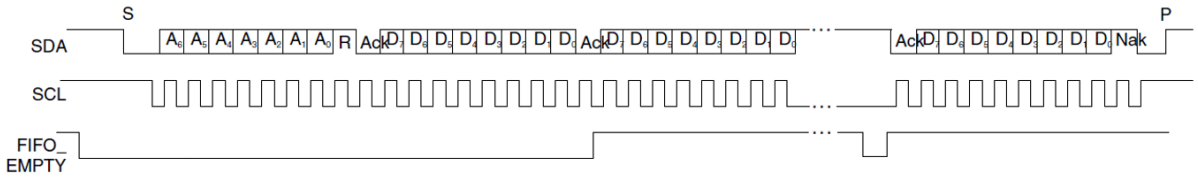


Рисунок 32.13. Поведение I²C, когда Tx FIFO становится пустым при работе в качестве мастер-приемника, генерация STOP

На Рисунок 32.14 и Рисунок 32.15 показаны конфигурации, в которых пользователь может управлять созданием условий RESTART на шине I²C. Если биты IC_DATA_CMD[10] и IC_RESTART_EN установлены, то RESTART генерируется до того, как байт данных будет записан или считан из слейва. Если функция перезапуска не включена, то сначала генерируется STOP, за которым следует START. На Рисунок 32.14 показана временная диаграмма, иллюстрирующая данную ситуацию при работе в качестве мастер-передатчика.

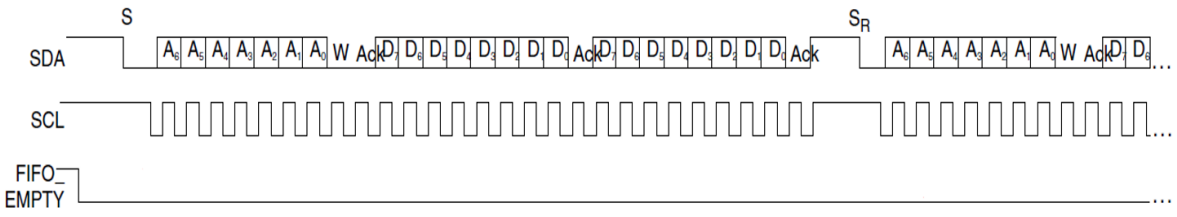


Рисунок 32.14. Мастер-передатчик, бит RESTART

Рисунок 32.15 иллюстрирует ту же ситуацию, но во время работы в качестве мастер-приемника.

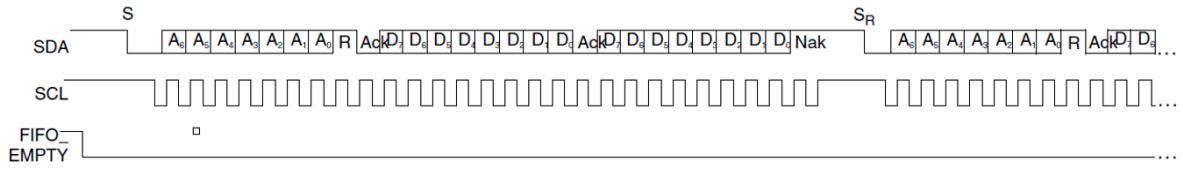


Рисунок 32.15. Мастер-приемник, бит RESTART

На Рисунок 32.16 показана временная диаграмма, иллюстрирующая поведение I²C, когда Tx FIFO не пустой и установлен STOP бит регистра IC_DATA_CMD при работе в качестве мастер-передатчика.

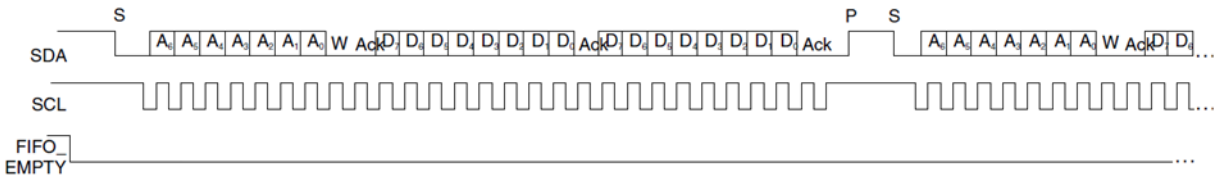


Рисунок 32.16. Мастер-передатчик, STOP бит в IC_DATA_CMD

Рисунок 32.17 иллюстрирует работу в качестве мастер-передатчика, когда первый байт, загруженный в Tx FIFO, пустой, а бит RESTART установлен.

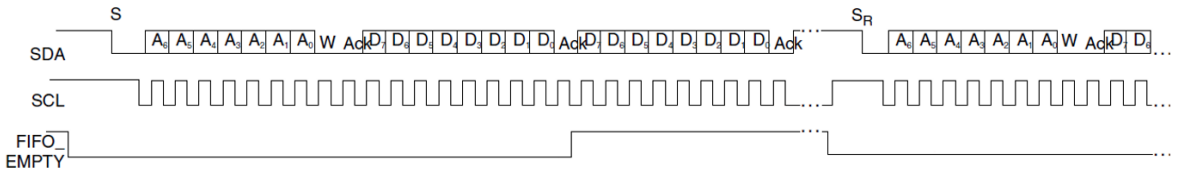


Рисунок 32.17. Мастер-передатчик, первый байт в Tx FIFO, бит RESTART

На Рисунок 32.18 показана временная диаграмма, иллюстрирующая поведение I^2C, когда Tx FIFO не пустой и установлен STOP бит регистра IC_DATA_CMD при работе в качестве мастер-приемника.

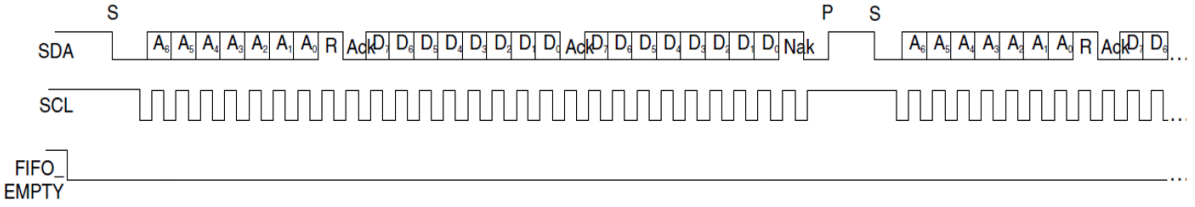


Рисунок 32.18. Мастер-приемник, STOP бит в IC_DATA_CMD

Рисунок 32.19 иллюстрирует работу в качестве мастер-приемника, когда первый байт, загруженный в Tx FIFO, пустой, а бит RESTART установлен.

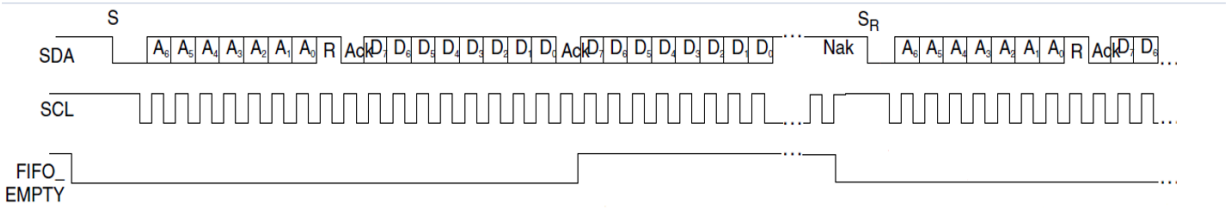


Рисунок 32.19. Мастер-приемник, первый байт в Tx FIFO, бит RESTART

32.6 Multi-master арбитраж

Возможно подключение нескольких мастеров к шине I^2C. Но если один мастер контролирует шину в конкретный момент времени, то никакое другой мастер не сможет получить контроль надо шиной, до тех пор, пока первый мастер не создаст на шине состояние IDLE, отправив последовательность Stop. При этом, возможна ситуация, при которой два или более мастеров пытаются одновременно получить контроль над шиной в состоянии IDLE, генерируя последовательность Start. В этом случае, они подвергаются арбитражу, структура которого представлена на Рисунок 32.20.

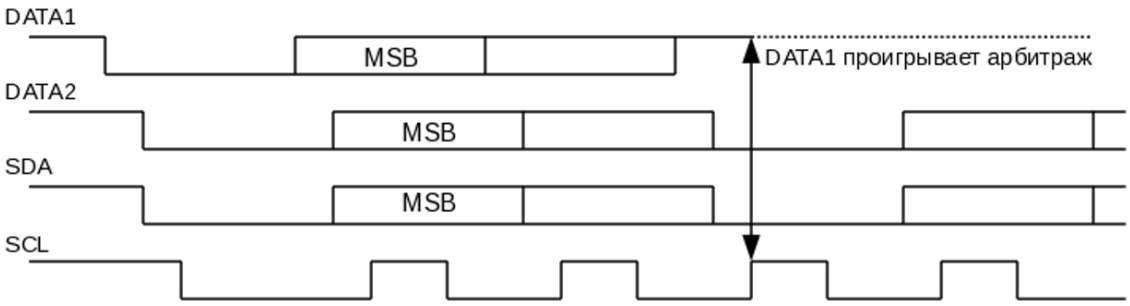


Рисунок 32.20. Арбитраж

Процедура арбитража начинается на линии SDA, когда линия SCL идет высоким уровнем. Мастер, который отправляет 1, когда другой мастер отправляет «0», проигрывает арбитраж и прекращает передачу данных. Мастер, проигравшей арбитраж, продолжает генерировать тактовый сигнал до конца текущего байта. Если оба мастера адресовали передачу одному и тому же слейву, то арбитраж продолжается в фазе передачи данных.

Синхронизация заключается в том, что во время арбитража тактовые сигналы мастеров складываются по «И» до окончания байта, на котором был завершен арбитраж.

Арбитаж не допускается между следующими условиями:

- Условие RESTART и бит данных;
- Условие STOP и бит данных;
- Условие RESTART и условие STOP.

Слейвы не учувствуют в процессе арбитража.

32.7 Синхронизация

Когда два или более мастера передают информацию по шине одновременно, они должны выполнить арбитраж и синхронизовать SCL. Все мастера генерируют свои собственные тактовые частоты для передачи сообщения. Данные действительный только во время высокого периода SCL. Когда мастер переводит SCL в 0, он начинает отсчет низкого времени (low time) SCL и переводит SCL в 1 в начале следующего тактового периода.

Однако, если другой мастер удерживает линию SCL в 0, то мастер переходит в состояние HIGH ожидания, пока SCL не перейдет в 1.

Затем все мастера отсчитывают свое высокое время (high time), и мастер с наименьшим высоким временем переводит SCL в 0. Затем мастера отсчитывают свое низкое время, и у кого низкое время выше, переводит другого мастера в состояние HIGH ожидания. Следовательно, генерируется SCL (Рисунок 32.21). Слейв устройства могут удерживать SCL на низком уровне, чтобы замедлить синхронизацию на шине I^2C.

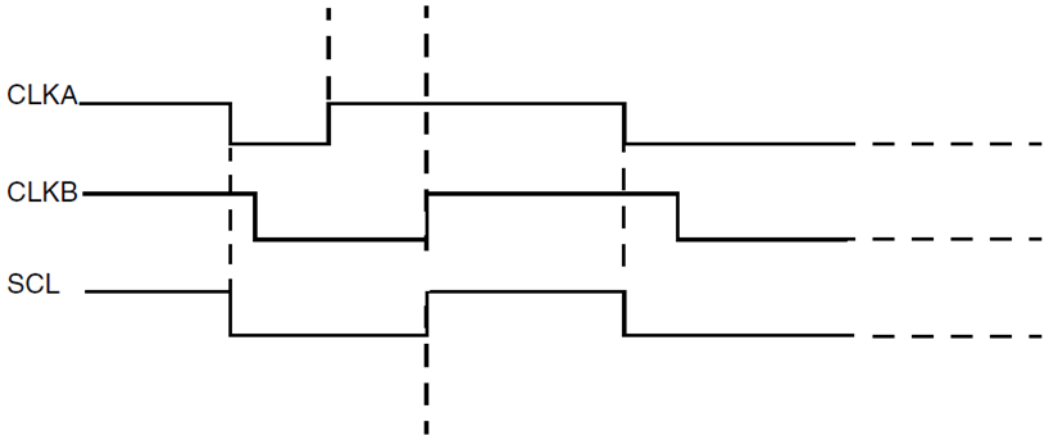


Рисунок 32.21. Синхронизация

32.8 Режимы работы

32.8.1 Слейв режим

32.8.1.1 Первоначальная конфигурация

Чтобы использовать I^2C в качестве слейва, нужно выполнить следующие условия:

- Отключить I^2C, записав 0 в регистр IC_ENABLE[0];
- В регистр IC_SAR[9:0] записать адрес слейва;
- В регистр IC_CON[3] записать тип адресации. Включить I^2C в режиме слейв, записав 0 в IC_SLAVE_DISABLE[6] и в MASTER_MODE[0];
- Включить I^2C, записав 1 в регистр IC_ENABLE[0].

32.8.1.2 Работоспособность слейв-передатчика для одного байта

Когда другой мастер I^2C на шине обращается к I^2C и запрашивает данные, I^2C действует, как слейв-передатчик, и выполняет следущие действия:

1. Другой мастер I^2C инициирует передачу I^2C с адресом, который совпадает со слейв адресом в IC_SAR.
2. I^2C подтверждает отправленный адрес и распознает направление передачи, указывая, что он работает, как слейв-передатчик.

3. I²C подтверждает прерывание RD_REQ и удерживает SCL на низком уровне. Он находится в состоянии ожидания, пока программное обеспечение не ответит.
4. Если есть оставшиеся данные в Tx FIFO до получения запроса на чтение, то I²C устанавливает прерывание TX_ABRT, чтобы очистить старые данные в Tx FIFO.
5. Программное обеспечение записывает в регистр IC_DATA_CMD данные для записи.
6. Программное обеспечение должно очистить прерывания RD_REQ и TX_ABRT.
7. I²C сбрасывает SCL и передает байт.
8. Мастер может удерживать шину I²C, выдав состояние RESTART, или освободить шину, выдав состояние STOP.

32.8.1.3 Работоспособность слейв-приемника для одного байта

Когда другой мастер I²C на шине обращается к I²C и отправляет данные, I²C действует, как слейв-приемник, и выполняет следущие действия:

1. Другой мастер I²C инициирует передачу I²C с адресом, который совпадает со слейв адресом в IC_SAR.
2. I²C подтверждает отправленный адрес и распознает направление передачи, указывая, что он работает, как слейв-приемник.
3. I²C принимает переданный байт и помещает его в буфер приема.
4. I²C подает сигнал прерывания RX_FULL.
5. Программное обеспечение может считывать байт из регистра IC_DATA_CMD.
6. Другой мастер может удерживать шину I²C, выдав состояние RESTART, или освободить шину, выдав состояние STOP.

32.8.1.4 Слейв-передатчик для массовых пересылок

I²C предназначен для обработки большого количества данных в TX FIFO, чтобы последующие запросы чтения могли принимать данные без прерываний, что позволяет уменьшить задержки.

Режим массовых пересылок возникает только тогда, когда I²C действует как слейв-передатчик. Если удаленный мастер подтверждает данные, отправленные слейв-передатчиком, и нет данных в TX FIFO слейва, I²C удерживает SCL на низком уровне, пока он вызывает RD_REQ и ожидает записи данных в TX FIFO перед оправкой удаленному мастеру.

Если RD_REQ замаскирован (M_RD_REQ = 0), то для активации периодических чтений регистра IC_RAW_INTR_STAT рекомендуется использовать процедуру синхронизации.

Прерывание RD_REQ возникает при запросе чтения и должно быть сброшено при выходе из процедуры обработки прерывания (ISR). ISR позволяет записать 1 или более 1 байта в Tx FIFO. Во время передачи байтов мастеру, слейв поднимаем RE_REQ, поскольку мастер запрашивает дополнительные данные.

Если программист заранее знает, что удаленный мастер запрашивает пакет из n байт, то, когда другой мастер обращается к I²C и запрашивает данные, в Tx FIFO может быть записано n байт, а удаленный мастер получает его как непрерывный поток данных.

Если удаленный мастер должен получить n байтов от I²C, но программист записал количество байтов, превышающее n, в Tx FIFO, то, когда слейв завершит отправку запрошенных n байтов, оно очищает Tx FIFO и игнорирует любые лишние байты.

I²C генерирует TX_ABRT для обозначения очистки Tx FIFO. В то время ожидается ACK/NACK. Если NACK получен, то удаленный мастер имеет все данные, которые ему нужны. В это время в конечном автомате слейва поднимается флаг, чтобы очистить оставшиеся данные в Tx FIFO. Этот флаг передается в область тактовой частоты шины процессора, где существует FIFO, и в это время очищается содержимое Tx FIFO.

32.8.2 Мастер режим

32.8.2.1 Первоначальная конфигурация

Начальная конфигурация зависит от параметра I2C_DYNAMIC_TAR_UPDATE.

32.8.2.1.1 I2C_DYNAMIC_TAR_UPDATE = 0

Чтобы использовать I²C в качестве мастера, выполните следующие действия:

1. Отключить I²C, записав 0 в IC_ENABLE[0];
2. Установить поддерживаемый режим максимальной скорости и желаемую скорость передачи. Убедиться, что 1 записан в IC_SLAVE_DISABLE[6] и в MASTER_MODE[0].
3. Записать в IC_TAR адрес устройства I²C. Это регистр также указывает, будет ли I²C выполнять общий вызов (General Call) или START BYTE.
4. Пункт для высокоскоростного режима. Записать в регистр IC_HS_MADDR мастер код для I²C (мастер код определяет программист).
5. Включить I²C, записав 1 в IC_ENABLE[0].
6. Записать направление передачи и данные для отправки в регистр IC_DATA_CMD. Если регистр IC_DATA_CMD записывается до включения I²C, данные и команды теряются.

32.8.2.1.2 I2C_DYNAMIC_TAR_UPDATE = 1

Чтобы использовать I²C в качестве мастера, выполните следующие действия:

1. Отключить I²C, записав 0 в IC_ENABLE[0];
2. Установить режим максимальной скорости, поддерживаемая для работы слейва, и указать, начинает ли I²C свои передачи в режиме адресации 7/10 бит, когда устройство является слейвом.

3. Записать в IC_TAR адрес устройства I²C, к которому нужно обратиться. Это регистр также указывает, будет ли I²C выполнять общий вызов (General Call) или START BYTE. Желаемая скорость передачи контролируется IC_10BITADDR_MASTER.
4. Пункт для высокоскоростного режима. Записать в регистр IC_HS_MADDR мастер код для I²C (мастер код определяет программист).
5. Включить I²C, записав 1 в IC_ENABLE[0].
6. Записать направление передачи и данные для отправки в регистр IC_DATA_CMD. Если регистр IC_DATA_CMD записывается до включения I²C, данные и команды теряются.

32.8.2.2 Динамическое обновление IC_TAR или IC_10BITADDR_MASTER

I²C поддерживает динамическое обновление IC_TAR[9:0] и IC_10BITADDR_MASTER (IC_TAR[12]). Чтобы выполнить динамическое обновление регистра IC_TAR, параметр I2C_DYNAMIC_TAR_UPDATE должен быть равен 1. Можно динамически записывать в регистр IC_TAR при условии, что программное обеспечение гарантирует, что в Tx FIFO нет других команд, которые используют существующий TAR адрес. Если данное условие не выполняется, то IC_TAR следует перепрограммировать только при следующих условий:

- IC_ENABLE[0] = 0 или

  IC_ENABLE[0] = 1 & IC_STATUS[5] = 0 & IC_CON[0] = 1 & IC_STATUS[2] = 1.

Вы можете динамически изменять адрес TAR, только если выполняются следующие условия:

- IC_ENABLE[0] = 1 & IC_EMPTYFIFO_HOLD_MASTER_EN = 1 & IC_CON[0] = 1 & IC_INTR_STAT[13]=1.

32.8.2.3 Передающий мастер и принимающий слейв

I²C поддерживает динамическое переключение между чтением и записью. Для передачи и записи данных, нужно записать данные в младший байт буфера данных I²C Rx/Tx и IC_DATA_CMD, CMD[8] = 0. Команда чтения записывает «don’t cares» в младший байт регистра IC_DATA_CMD, CMD[8] = 1. Мастер I²C продолжает передачу, пока есть команды в передающем FIFO. Если передающее FIFO становится пустым, мастер либо поднимает состояние STOP, либо проверяет, установлена ли 1 в IC_DATA_CMD[9]:

- 1 — выдает состояние STOP;
- 0 — SCL удерживается на низком уровне до тех пор, пока следующая команда не будет записана в передающее FIFO.

32.8.3 Отключение I²C

Регистр IC_ENABLE_STATUS позволяет программному обеспечению определить, когда аппаратное обеспечение полностью отключилось. Если программное обеспечение или

приложение знают, что I²C не использует адреса TAR для ожидающих команд в Tx FIFO, то можно обновить адреса TAR, даже если IC_STATUS[2] = 0.

32.8.3.1 Порядок действий

1. Определить ti2c_poll.
2. Определить MAX_T_POLL_COUNT.
3. Программным обеспечением предотвратить запуск основных транзакций, но позволять завершить любые ожидающие передачи.
4. POLL_COUNT инициализируется 0.
5. IC_ENABLE[0] = 0.
6. Прочитать IC_ENABLE_STATUS и проверить IC_EN[0]. Увеличить POLL_COUNT на 1. Если POLL_COUNT >= MAX_T_POLL_COUNT, выйти с ошибкой.
7. Если IC_ENABLE_STATUS[0] = 1, переход к предыдущему шагу.

32.8.4 Прерывание передачи I²C

Бит ABORT в регистре IC_ENABLE позволяет программному обеспечению освободить шину I²C до завершения выданных команд передачи из Tx FIFO. В ответ на запрос ABORT контроллер выдает состояние STOP по шине I²C, за которым следует сброс Tx FIFO. Прерывание передачи разрешено только в режиме работы мастера.

32.8.4.1 Порядок действий

1. Прекратить заполнять Tx FIFO (IC_DATA_CMD) новыми командами.
2. Режим DMA. Отключить передачи DMA (TDMAE = 0).
3. IC_ENABLE[1] = 1.
4. Дождаться прерывания M_TX_ABRT.
5. Прочитать регистр IC_TX_ABORT_SOURCE, чтобы определить источник как ABORT_USER_ABORT.

32.9 Фильтрация

Фильтрующая логика основана на счетчиках, которые отслеживают состояние входных сигналов (SDA и SCL). Запуск счетчика происходит, когда соответствующий ему сигнал меняет значение. Сигнал фиксируется контроллером, когда его длительность становится равна установленному в регистре IC_*_SPLKEN числу тактов фильтрующего счетчика. Принцип работы механизма фильтрации представлен на Рисунок 32.22.

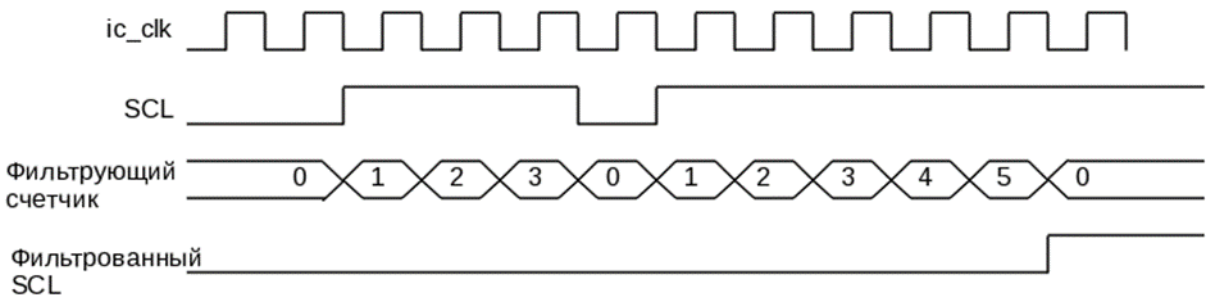


Рисунок 32.22. Фильтрация

Спецификация шины I^2C требует различной максимальной длины импульсов в зависимости от режима работы – 50 нс для SS и FS, 10 нс для HS и UFm. Поэтому для хранения значений, необходимых для каждого случая, требуется три регистра:

- Регистр IC_FS_SPKLEN для SS и FS.
- Регистр IC_HS_SPKLEN для HS.
- Регистр IC_UFM_SPKLEN для UFm.

Данные регистры имеют ширину 8 бит и доступны через интерфейс APB для чтения и записи. Однако, они могут быть записаны только тогда, когда I^2C отключен.

32.10 Плюс-быстрый режим работы (Fm+)

В Fm+ I^2C поддерживает скорость до 1000 Кбит/с. Чтобы включить Fm+, нужно выполнить следующие шаги перед началом любой передачи данных:

1. Настроить мастер или слейв I^2C на быстрый (Fm) или высокоскоростной (Hm) режим (IC_MAX_SPEED_MODE> = 2).
2. Установить ic_clk больше или равную 32 МГц.
3. Запрограммировать регистр IC_CON[2:1] = 2’b10 для Fm или Fm+.
4. Запрограммируйте регистры IC_FS_SCL_LCD и IC_FS_SCL_HCL, чтобы соответствовать Fm+ SCL.
5. Запрограммируйте регистр IC_FS_SPKLEN на подавление максимального всплеска 50 нс.
6. Запрограммируйте регистр IC_SDA_SETUP, чтобы обеспечить минимальное время установки данных.

32.11 Функция очистки шины

I^2C поддерживает функцию очистки шины, которая обеспечивает плавное восстановление SDA и SCL во время маловероятных событий, когда тактовая линия или линия данных застревает на низком уровне.

32.11.1 SDA на низком уровне

Если SDA застряла на низком уровне, мастер выполняет следующие действия (рисунок 32. 23 и рисунок 32. 24):

1. Мастер отправляет максимум 9 тактовых импульсов, чтобы восстановить низкий уровень шины в течение этих 9 тактов.
2. Если линия SDA восстанавливается в течение 9 тактовых импульсов, тогда мастер отправит STOP, чтобы освободить шину.
3. Если линия SDA не восстанавливается даже после 9-го тактового импульса, системе требуется аппаратный сброс.

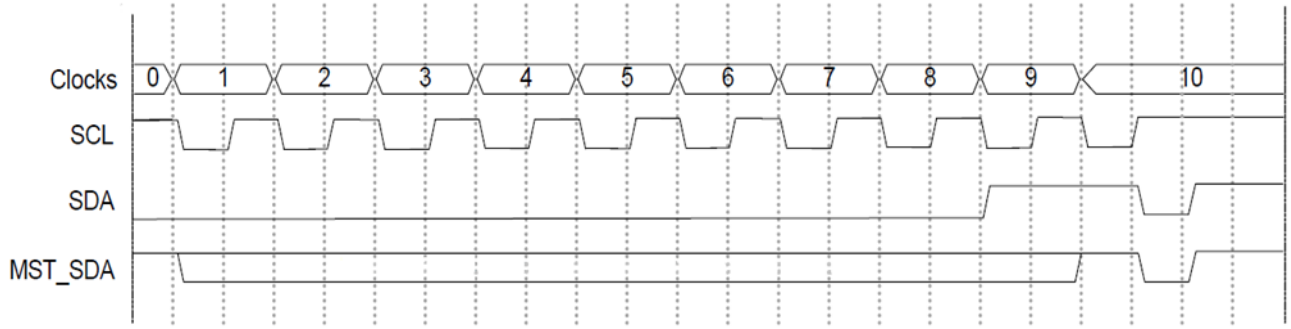


Рисунок 32.23. Восстановление SDA (9 тактов SCL)

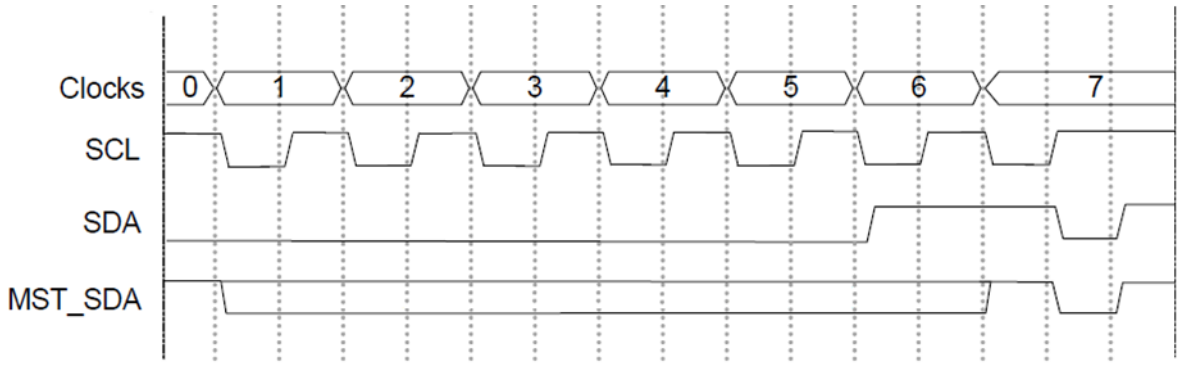


Рисунок 32.24. Восстановление SDA (6 тактов SCL)

32.11.2 SCL на низком уровне

Когда SCL застревает на низком уровне, требуется сбросить шину с помощью аппаратного сброса.

32.12 Режим сверхбыстрой скорости (UFm)

UFm —режим скорости шины I^2C, который передает данные в одном направлении.

UFm основан на стандартом протоколе I²C. Командный бит всегда должен иметь значение «запись» только потому, что это однонаправленная шина. Бит данных в ACK устанавливается мастером в 1, игнорирую цикл ACK. Для UFm используется двухтактный драйвер.

Мастер состоит из выходных сигналов ic_clk_oe, USCL и ic_data_oe, USDA. Выходные сигналы имеют активный низкий уровень.

Слейв состоит из входных сигналов ic_clk_in_a, USCL и ic_data_in_a, USDA. Входные сигналы имеют активный высокий уровень.

Шина UFm не поддерживает работу с несколькими мастерами. В шине UFm мастер является единственным устройством, который инициирует передачу данных по шине и предоставляет тактовые сигналы для поддержки этой передачи. Все остальные устройства являются слейвами.

Формат байта, генерация START и STOP такие же, как и в других режимах I²C, за исключением игнорирования цикла ACK. Слейв никогда ничего не управляет на шине, мастер всегда управляет NACK в течение ACK передачи (Рисунок 32.25).

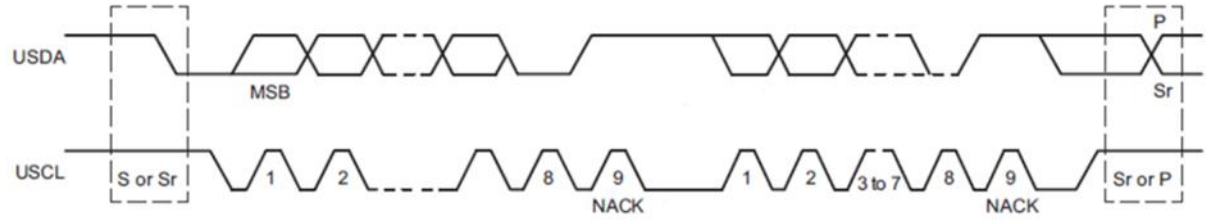


Рисунок 32.25. Передача байтов в Ufm

Комбинированный формат протокола I²C не поддерживается в режиме UFm. 10 – битная адресация поддерживается в режиме UFm и ведет себя аналогично другим режимам (Рисунок 32.26).



Рисунок 32.26. Передача записи при 10 – битной адресации

Режим UFm поддерживает функции START и общий вызов, аналогичным другим режимам I²C. Если слейв не отвечает, то мастер может выполнить сброс с помощью программного или аппаратного сброса.

32.13 SMBus/PMBus

Протокол SMBus разработан для создания предсказуемой линии соединения. Протокол определяет для устройств набор таймаутов и условий их возникновения.

32.13.1 tTIMOUT,MIN

Таймаут tTIMOUT,MIN позволяет мастеру или слейву детектировать нарушение обмена данными с неисправным девайсом, в случае, когда этот девайс слишком долго удерживает линию SMBCLK на низком уровне. От любого слейва на шине требуется, чтобы оно отпускало шину (перестало управлять сигналами, и позволило SMBCLK и SMBDAT перейти в 1) в случае, если SMBCLK удерживается в 1 хотя бы на 1 такт дольше tTIMEOUT,MIN. Устройство, обнаружившее это состояние, должно сбросить интерфейс и быть готовым принять новую START-последовательность не позднее tTIMEOUT,MAX.

Контроллер I^2C включает возможность сброса шины в режиме SMBus. Пользователь может использовать регистр IC_SCL_STUCK_TIMEOUT для настройки параметра tTIMEOUT,MIN.

Контроллер I^2C в режиме слейва прекратит коммуникацию по интерфейсу и отпустит обе линии SCL и SDA после установки прерывания SCL_STUCK_TIMEOUT.

Контроллер I^2C в режиме мастер имеет возможность инициализировать состояние Abort. В этом случае передача данных прекращается и генерируется STOP-последовательность на шине. Управление фичей осуществляется программно через бит IC_ENABLE[1].

32.13.2 Master Device Clock Extension

Интервал tLow:MEXT определяется как совокупное время, в процессе которого мастер может расширять такты в пределах одного байта в сообщении от:

- START до ACK
- ACK до ACK
- ACK до STOP

Контроллер I^2C в режиме мастер использует регистр IC_SMBUS_CLOCK_LOW_MEXT для детектирования таймаута на расширение клока и генерации прерывания SMBUS_CLK_LOW_MEXT.

32.13.3 Slave Device Clock Extension

Интервал tLOW:SEXT - это совокупное время, в течении которого слейв может расширять такты частоты внутри одного сообщения от START до STOP.

Контроллер в режиме мастер использует регистр IC_SMBUS_CLOCK_LOW_SEXT для детектирования расширения таймаута клока от слейва и генерации прерывания SMBUS_CLK_LOW_SEXT.

Мастер может прервать обмен данными с любым слейвом, который нарушает спецификацию на tLOW:SEXT или tTIMEOUT,MIN. Управление осуществляется через бит IC_ENABLE[1].

32.13.4 Таймаут SMBDAT-Low

Неисправный агент может удерживать линию SMBDAT в низком уровне бесконечно. Это не позволит мастеру отправить последовательность STOP и закончить передачу. Если SMBUSDAT находится в низком уровне в течении tTIMEOUT,MAX после перехода в 1 линии SMBCLK в конце транзакции, мастер будет удерживать SMBCLK в 0 как минимум tTIMEOUT,MAX в попытке перезапустить SMBus-интерфейс для всех устройств на шине.

Пользователь может использовать регистр IC_SDA_STUCK_TIMEOUT для управления обнаружением тайматуа SMBDAT-Low. Если происходит залипание линии SMBDAT на низком уровне, генерируется состояние SDA_STUCK_TIMEOUT, после чего необходимо программно установить бит SMBUS_CLK_RESET регистра IC_ENABLE для удержания SCL на низком уровне. Это приведет к возникновению таймаута IC_SCL_STUCK_TIMEOUT, который приведет к сбросу SMBus-интерфейса для всех девайсов на шине.

32.13.5 Bus Protocol

В режиме SMBus, контроллер поддерживает набор команд для осуществления передачи данных. Все команды имеют длину в 1 байт, в то время как размер их аргументов и возвращаемых значений может варьироваться. В соответствии со спецификацией SMBus, передача данных начинается с наиболее значимого бита (MSB). Протокол содержит 11 команд: Quick Command, Send Byte, Receive Byte, Write Byte, Write Word, Read Byte, Read Word, Process Call, Block Read, Block Write, Block Write – Block Read, Process Call.

Протокол SMBs предназначен для передачи сообщений, которые в общем виде отличаются от команд передачи данных I^2C. Однако, существует возможность настроить SMBus-мастер на передачу I^2C команд. Следующая таблица объясняет образование команд SMBus через Tx-FIFO команды контроллера.

В режиме SMBus-мастер, все принятые байты данных будут доступны в Rx_FIFO. В режиме SMBus-слейв все команды протокола и байты данных будут приняты в Rx-FIFO и запрос на чтение байт данных должен быть отправлен через Tx-FIFO, аналогично режиму I^2C.

Таблица 32.60. Протоколы SMBus

Protocol	Required Tx FIFO Commants	Command/Data (IC_DATA_CMD[7:0])	CMD bit (IC_DATA_CMD[8])	STOP bit (IC_DATA_CMD[9])	Remarks
Quick Command	1	Not Applicable	Set the command [R/W]	Set to 1	Set IC_TAR[11] and IC_TAR[16] to 1
Send Byte	1	Data Byte	Set to 0	Set to 1	
Receive Byte	1	Not Applicable	Set to 1	Set to 1	
Write Byte	2	Command Code	Set to 0	Set to 0	
		Data Byte	Set to 0	Set to 1	
Write Word	3	Command Code	Set to 0	Set to 0	
		Data Byte Low	Set to 0	Set to 0	
		Data Byte High	Set to 0	Set to 1	
Read Byte	2	Command Code	Set to 0	Set to 0	
		Not Applicable	Set to 1	Set to 1	
Read Word	3	Command Code	Set to 0	Set to 0	
		Not Applicable	Set to 1	Set to 0	
		Not Applicable	Set to 1	Set to 1	
Process Call	5	Command Code	Set to 0	Set to 0	
		Data Byte Low	Set to 0	Set to 0	
		Data Byte High	Set to 0	Set to 0	
		Not Applicable	Set to 1	Set to 0	
		Not Applicable	Set to 1	Set to 1	
Block Write	N+2	Command Code	Set to 0	Set to 0	
		Data Byte	Set to 0	Set to 0	
		N+1) Data Byte N	Set to 0	Set to 1	
Block Read	N+2	Command Code	Set to 0	Set to 0	
		Not Applicable	Set to 0	Set to 0	
		N+1) Not Applicable	Set to 0	Set to 1	
Block Write Block Read Process Call	M+N+2	Command Code	Set to 0	Set to 0	
		Data Byte 1	Set to 0	Set to 0	
		M+1) Data Byte M	Set to 0	Set to 0	
		M+2) Not Applicable	Set to 1	Set to 0	
		M+3) Not Applicable	Set to 1	Set to 0	
		M+N+1) Not Applicable	Set to 1	Set to 1	
SMBUS Host Notify Protocol	3	Device-Address	Set to 0	Set to 0	Set IC_TAR[6:0] to SMB Host
		Data Byte Low	Set to 0	Set to 0	

Protocol	Required Tx FIFO Commants	Command/Data (IC_DATA_CMD[7:0])	CMD bit (IC_DATA_CMD[8])	STOP bit (IC_DATA_CMD[9])	Remarks
		Data Byte High	Set to 0	Set to 1	Address (0001 000)

I²C слейву можно разрешить принимать только быстрые команды путем включения бита SLAVE_QUICK_CMD_EN. Слейв I²C выдает прерывание SMBUS_QUICK_DET при получении команды QUICK.

SMBus вводит механизм проверки ошибок пакетов путем добавления байта PEC в конце протокола шины.

32.13.6 SMBUS Address Resolution Protocol

Конфликты адреса слейва SMBus могут быть разрешены путем динамического присвоения нового уникального адреса каждому слейву. Эта функция позволяет подключать устройства к системе в «hot-plugged» режиме.

SMBus вводит 128 – битный UDID для каждого устройства в системе. I²C использует параметр IC_SMBUS_UDID_MSB (96 бит) и IC_SMBUS_ARP_UDID_LSB (32 бит) в UDID.

I²C использует бит PERSISTANT_SLV_ADDR_EN в регистре IC_CON, чтобы указать, поддерживает ли I²C постоянный адрес слейва.

Ведущее устройство I²C может выдавать общие и направленные команды ARP для назначения динамического адреса слейв устройствам в системе SMBus.

Таблица 32.61 описывает вывод команд SMBus ARP через команды Tx-FIFO в I²C.

Таблица 32.61. Вывод команд SMBus ARP

ARP Command	Required Tx FIFO Commants	Command/Data (IC_DATA_CMD[7:0])	CMD bit (IC_DATA_CMD[8])	STOP bit (IC_DATA_CMD[9])	Remarks
Prepare for ARP	2	Command =’0000 0001’	Set to 0	Set to 0	Set IC_TAR[6:0] to SMB Default Address(1100 001)
		PEC Byte	Set to 0	Set to 1	
Reset Device (General)	2	Command =’0000 0010’	Set to 0	Set to 0	Set IC_TAR[6:0] to SMB Default Address(1100 001)
		PEC Byte	Set to 0	Set to 1	
Get UDID (General)	20	Command =’0000 0011’	Set to 0	Set to 0	1. Set IC_TAR[6:0] to SMB Default Address(1100 001).
		Not Application	Set to 1	Set to 0	
		Not Application	Set to 1	Set to 0	
		Not Application	Set to 1	Set to 0	

ARP Command	Required Tx FIFO Commants	Command/Data (IC_DATA_CMD[7:0])	CMD bit (IC_DATA_CMD[8])	STOP bit (IC_DATA_CMD[9])	Remarks
		PEC Byte	Set to 1	Set to 1	2.16 Reads to be performed for the 128 UDID bytes. 3.Last read command for the slave address.
Assign Address	20	Command =’0000 0011’	Set to 0	Set to 0	1. Set IC_TAR[6:0] to SMB Default Address(1100 001). 2.16 Reads to be performed for the 128 UDID byte. 3.Last Write command for the Assigned slave address.
		Byte Count = 17	Set to 0	Set to 0	
		UDID Byte 15	Set to 0	Set to 0	
		UDID Byte 14	Set to 0	Set to 0	
		Assign Address	Set to 0	Set to 0	
		PEC Byte	Set to 1	Set to 1	
Get UDID (Directed)	19	Command =’0000_0011’	Set to 0	Set to 0	1. Set IC_TAR[6:0] to SMB Default Address(1100 001). 2.16 Reads to be performed for the 128 UDID byte. 3.Last Read command for the slave address.
		Slave address[6:0],1}	Set to 1	Set to 0	
		Not Applicable	Set to 1	Set to 0	
		Not Applicable	Set to 1	Set to 0	
		PEC Byte	Set to 1	Set to 1	
Reset Device (Directed)	2	Command = {slave address[6:0], 0}	Set to 0	Set to 0	Set IC_TAR[6:0] to SMB Default Address(1100 001).
		PEC byte	Set to 0	Set to 1	
Notify ARP Master	3	Device Address = ‘1100 0010’	Set to 0	Set to 0	Set IC_TAR[6:0] to SMB Host Address(0001 000).
		Data Byte Low = ‘0000 0000’	Set to 0	Set to 0	
		Data Byte High = ‘0000 0000’	Set to 0	Set to 1	

32.13.7 SMBUS Additional Slave Address

I^2C поддерживает вторую дополнительную возможность декодирование адреса ведомого устройства. Он может быть сконфигурирован так, чтобы содержать дополнительный регистр адреса слейва IC_OPTIONAL_SAR. Если данный регистр настроен, пользователь может записать в него любой действительный адрес слейва, который будет совпадать с

входящим адресом слейва на SMBus. Совпадение входящего адреса с регистром IC_SAR или IC_OPTION_SAR приведет к тому, что I^2C подтвердит транзакцию и ответит на нее. Данный регистр контролируется битом OPTIONAL_SAR_CTRL. Если OPTIONAL_SAR_CTRL равен 1, то IC_OPTIONAL_SAR будет использоваться для сопоставления входящего адреса. Все ограничения регистра IC_SAR распространяются и на регистр IC_OPTIONAL_SAR.

Значение по умолчанию, которое регистр IC_OPTIONAL_SAR получает после сброса, может быть настроено параметром IC_OPTIONAL_SET_DEFAULT.

32.13.8 SMBUS Optional Signals

Стандарт SMBus поддерживает дополнительные сигналы:

- SMBus Suspend Signal
- SMBus Alert Signal

Значение по умолчанию, которое регистр IC_OPTIONAL_SAR получает после сброса, можно настроить с помощью параметра IC_OPTIONAL_SAR_DEFAULT.

32.13.8.1 SMBus Suspend Signal (SMBSUS)

SMBSUS — это дополнительный сигнал, который подается системным контроллером, чтобы указать, что система должна перейти в режим ожидания с низким энергопотреблением. SMBSUS является выходом из системного контроллера и входом для всех остальных устройств. SMBSUS имеет активный низкий сигнал. I^2C реализует функцию, используя следующие сигналы:

- ic_smbsus_in_n;
- ic_smbsus_out_n.

Выходной сигнал ic_smbsus_out_n управляется битом SMBUS_SUS_CTRL. Если SMBUS_SUS_CTRL равен 1, то сигнал ic_smbsus_out_n переходит в 0, как только мастер завершит любую текущую передачу. Для выхода из режима ожидания, пользователю необходимо сбросить этот бит.

Входной сигнал ic_smbsus_in_n генерирует прерывание ic_smbsus_det_intr (или ic_smbsus_det_intr_n) по заднему фронту. Программное обеспечение может использовать это прерывание для перехода в режим с низким употреблением. Текущее состояние ic_smbsus_in_n можно узнать из бита SMBUS_SUSPEND_STATUS.

32.13.8.2 SMBus Alert Signal (SMBALERT)

SMBALERT — это другой дополнительный сигнал, определенный стандартом SMBus. Сигнал является входным для главного устройства и выходным для всех остальных устройств. Поскольку несколько устройств могут реализовать SMBALERT, он должен быть

сигналов типа wire-AND. Обнаружив сигнал SMBALERT, хост должен отправить Alert Response Address. Он отправляет адрес хосту и отменяет Alert Signal. Если хост по-прежнему обнаруживает Alert Signal, он повторяет отправку Alert Response Address. I²C реализует функцию, используя следующие сигналы:

- ic_smbalert_in_n;
- ic_smbalert_oe.

Выходной сигнал ic_smbalert_oe используется аналогично ic_clk_oe и ic_data_oe и контролируется битом SMBUS_ALERT_CTRL. После подтверждения пользователем, I²C ждет, пока мастер не отправит адрес ответа на предупреждение. Получив его, содержимое регистра IC_SAR[7:0] отправляется мастеру. В случае успеха, I²C очищает бит SMBUS_ALERT_CTRL и отменяет утверждение сигнала ic_smbalert_oe.

Входной сигнал ic_smbalert_in_n генерирует прерывание ic_smbalert_det_intr (или ic_smbalert_det_intr_n) по заднему фронту. При работе в качестве хоста пользователю необходимо ответить на прерывание, отправив команду чтения байта с Alter Response Address. Текущий статус ic_smbalert_in_n можно прочитать из бита SMBUS_ALERT_STATUS.

32.14 Настройка частоты IC_CLK

Контроллер I²C содержит следующие регистры при режимах работы SS, FS, FM+ и HS:

- IC_SS_SCL_HCNT;
- IC_SS_SCL_LCNT;
- IC_FS_SCL_HCNT;
- IC_FS_SCL_LCNT;
- IC_HS_SCL_HCNT;
- IC_HS_SCL_LCNT.

Контроллер I²C содержит следующие регистры при режиме работы UFm:

- IC_UFM_SCL_HCNT;
- IC_UFM_SCL_LCNT.

Таблица 32.62. Параметры синхронизации I²C из регистров *CNT

Timing Parameter	Symbol	Standard Speed	Fast Speed / Fast Speed Plus	High Speed (100 pf)	High Speed (400 pf)
LOW period of the SCL clock	tLow	IC_SS_SCL_LCNT	IC_FS_SCL_LCNT	IC_HS_SCL_LCNT	IC_HS_SCL_LCNT

Timing Parameter	Symbol	Standard Speed	Fast Speed / Fast Speed Plus	High Speed (100 pf)	High Speed (400 pf)
HIGH period of the SCL clock	tHIGH	IC_SS_SCL_HCNT	IC_FS_SCL_HCNT	IC_HS_SCL_HCNT	IC_HS_SCL_HCNT
Setup time for a repeat START condition	tSU; STA	IC_SS_SCL_LCNT	IC_FS_SCL_HCNT	IC_HS_SCL_LCNT	(IC_HS_SCL_LCNT) / 2
Hold time (repeated) START condition*	tHD; STA	IC_SS_SCL_HCNT	IC_FS_SCL_HCNT	IC_HS_SCL_LCNT	(IC_HS_SCL_LCNT) / 2
Setup time for STOP condition	tSU; STO	IC_SS_SCL_HCNT	IC_FS_SCL_HCNT	IC_HS_SCL_LCNT	(IC_HS_SCL_LCNT) / 2
Bus free time between a STOP and a START condition	tBUF	IC_SS_SCL_LCNT	IC_FS_SCL_LCNT	NA	NA
Spike length	tSP	IC_FS_SPKLEN	IC_FS_SPKLEN	IC_HS_SPKLEN	IC_HS_SPKLEN
Data hold time	tHD; DAT	IC_SDA_HOLD	IC_SDA_HOLD	IC_SDA_HOLD	IC_SDA_HOLD
Data setup time	tSU; DAT	IC_SDA_SETUP	IC_SDA_SETUP	IC_SDA_SETUP	IC_SDA_SETUP

32.14.1 Minimum High and Low Count in SS, FS, FM+ and HS Modes with IC_CLK_FREQ_OPTIMIZATION = 0

Когда контроллер работает в режиме мастер, должны выполняться следующие условия:

- значения регистров IC_SS_SCL_LCNT и IC_FS_SCL_LCNT должны быть больше, чем IC_FS_SPKLEN+7;
- значения регистров IC_SS_SCL_HCNT и IC_FS_SCL_HCNT должны быть больше, чем IC_FS_SPKLEN+5;
- значение регистра IC_HS_SCL_LCNT должно быть выше, чем IC_HS_SPKLEN+7;
- значение регистра IC_HS_SCL_HCNT должно быть выше, чем IC_HS_SPKLEN+5.

Подробная информация о high и low count:

- Минимальное значение IC_*_ SPKLEN+7 для регистров *_LCNT связано со временем, необходимым I^2C для управления SDA после отрицательного фронта SCL.
- Минимальное значение IC_*_ SPKLEN+5 для регистров *_HCNT связано со временем, необходимым I^2C для выборки SDA в течение высокого периода SCL.

- I²C добавляет один цикл к запрограммированному значению *_LCNT, чтобы сгенерировать низкий период тактового сигнала SCL.
- I²C добавляет IC_*_ SPKLEN+7 циклов к запрограммированному значению *_HCNT, чтобы сгенерировать высокий период тактового сигнала SCL

Временные параметры обмена представлены на Рисунок 32.27.

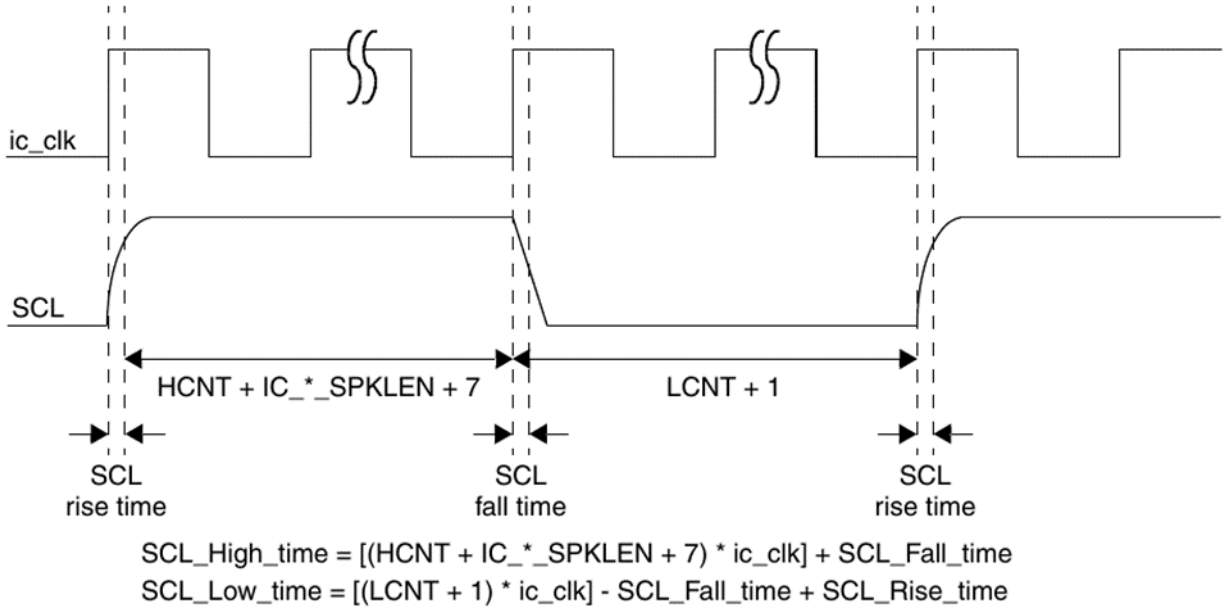


Рисунок 32.27. Временные параметры обмена

32.14.2 Minimum High and Low Count in SS, FS, FM+ and HS Modes with IC_CLK_FREQ_OPTIMIZATION = 1

- SCL LOW, управляемый I²C, будет равен значению регистра IC_*_LCNT. Аппаратное обеспечение не поддерживает запись в IC_*_LCNT значения меньше 6. Минимальное время SCL LOW составляет 6 периодов ic_clk.
- SCL HIGH, управляемый I²C, будет равен значению регистра IC_*_HCNT + SPKLEN + 3.

Общее время высокого и низкого уровня SCL зависит от времени нарастания и спада SCL. Параметры времени нарастания и спада SCL зависят от внешних факторов:

- Характеристики ввода-вывода драйвера;
- Значение нагрузочного резистора;
- Общая емкость на линии SCL и т.д.

I²C не может контролировать эти характеристики.

32.14.3 Minimum High and Low count in Ultra-Fast mode (IC_ULTRA_FAST_MODE = 1)

Когда I²C работает как мастер:

- Значение регистра IC_UFM_SCL_HCNT должно быть больше или равно 3.
- Значение регистра IC_UFM_SCL_LCNT должно быть больше или равно 5.

32.15 SDA Hold Time

Спецификация протокола I²C требует 300 нс времени удержания сигнала SDA (tHD; DAT) в SS и Fm.

Задержки на сигналах SCL и SDA могут означать, что требование о времени удержания выполняется на мастере, но не на слейве, или наоборот. Поскольку каждое приложение сталкивается с различными задержками, I²C содержит программируемый регистр (IC_SDA_HOLD), позволяющий динамически регулировать время удержания SDA.

IC_SDA_HOLD[15:0] используются для управления временем удержания SDA во время передачи как слейву, так и мастеру (после перехода SCL с HIGH в LOW).

IC_SDA_HOLD[23:16] используются для продления перехода SDA всякий раз, когда уровень SCL в приемнике HIGH.

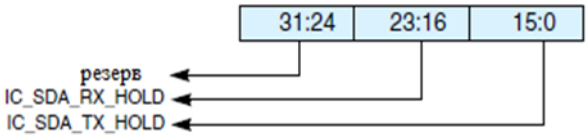


Рисунок 32.28. Регистр IC_SDA_HOLD

Если для разных скоростных режимов требуется разное время удержания SDA, то регистр IC_SDA_HOLD должен быть перепрограммирован при изменении скоростного режима. Регистр IC_SDA_HOLD может быть запрограммирован только в том, случае, если I²C отключен.

32.15.1 SDA Hold Timing in Receiver

Когда I²C действует как приемник, согласно протоколу I²C, устройство должно удерживать линию SDA, чтобы преодолеть неопределенный разрыв между 1 и 0 SCL.

IC_SDA_RX_HOLD можно использовать для изменения внутреннего времени удержания, которое I²C применяет к SDA. Каждое значение в регистре IC_SDA_RX_HOLD представляет собой единицу измерения одного периода ic_clk. Минимальное значение

IC_SDA_RX_HOLD равно 0. Это время удержания применимо только при HIGH SCL. Приемник не расширяет SDA после того, как SCL становится LOW.

На Рисунок 32.29 показан приемник I^2C с запрограммированным значением IC_SDA_RX_HOLD, большим или равным 3.

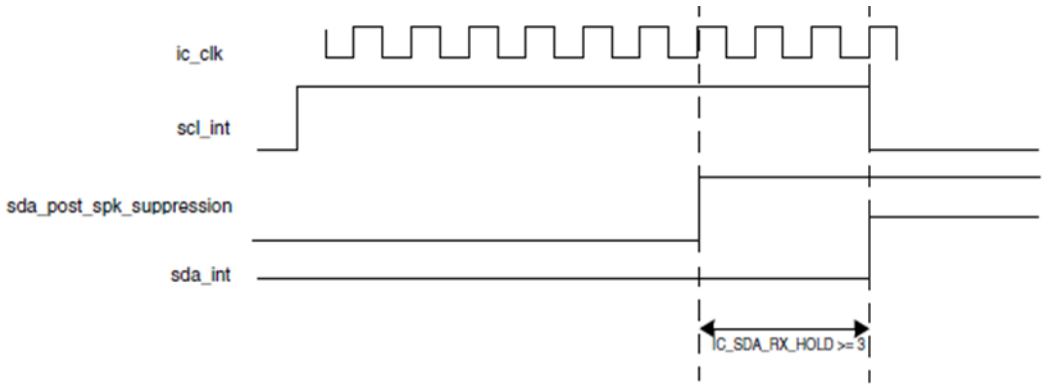


Рисунок 32.29. IC_SDA_RX_HOLD >= 3

Если IC_SDA_RX_HOLD больше 3, I^2C не удерживает SDA более 3 циклов ic_clk, потому что SCL внутренне переходит в LOW.

Рисунок 32.30 показывает I^2C, как приемник с IC_SDA_RX_HOLD = 2.

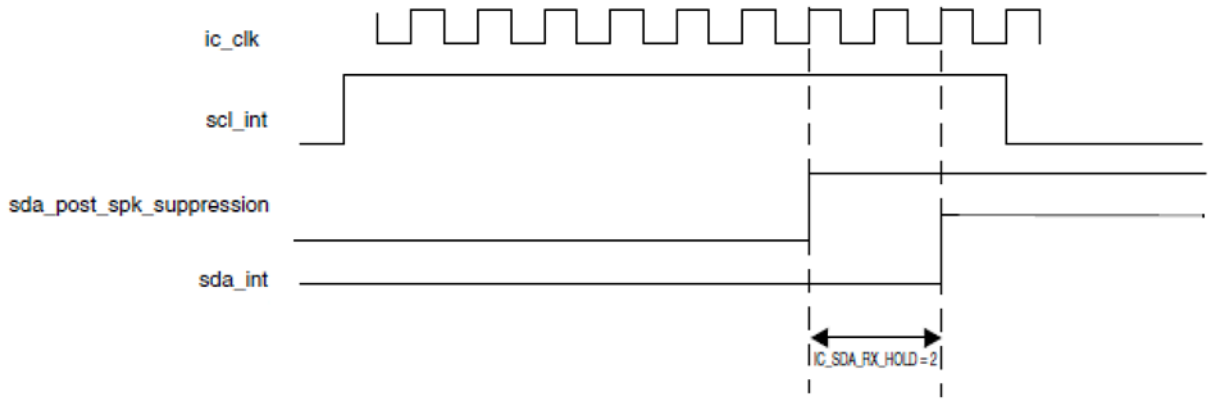


Рисунок 32.30. IC_SDA_RX_HOLD = 2

Максимальные значения IC_SDA_RX_HOLD, которые могут быть запрограммированы в регистре для соответствующих скоростных режимов, приведенных в Таблица 32.63.

Таблица 32.63. Максимальные значения IC_SDA_RX_HOLD

Speed Mode	Maximum IC_SDA_RX_HOLD Value
Standard Mode	IC_SS_SCL_HCNT – IC_FS_SPKLEN – 3
Fast Mode or Fast Node Plus	IC_SS_SCL_HCNT – IC_FS_SPKLEN – 3
High Speed (IC_CAP_LOADING = 100)	Min{ IC_FS_SCL_HCNT – IC_FS_SPKLEN – 3, IC_HS_SCL_LCNT – IC_HS_SPKLEN - 3}
High Speed (IC_CAP_LOADING = 400)	Min{ IC_FS_SCL_HCNT – IC_FS_SPKLEN – 3, (IC_HS_SCL_LCNT/2) – IC_HS_SPKLEN - 3}

32.15.2 SDA Hold Timing in Transmitter

Регистр IC_SDA_TX_HOLD может использоваться для изменения синхронизации сгенерированного сигнала SDA (ic_data_oe) с помощью I^2C. Каждое значение в регистре IC_SDA_TX_HOLD представляет собой единицу измерения одного периода ic_clk.

Когда I^2C работает в режиме мастер, минимальное время THD:DAT составляет один период ic_clk. Следовательно, даже, если значение IC_SDA_TX_HOLD равно нулю, I^2C приведет SDA (ic_data_oe) к одному циклу ic_clk после приведения SCL (ic_clk_oe) к логическому 0.

Когда I^2C работает в слейв режиме, минимальное время THD:DAT составляет SPKLEN+7 периодов ic_clk, где SPKLEN равно:

- IC_FS_SPKLEN, при SS, Fm или Fm+;
- IC_HS_SPKLEN, при Hm.

Эта задержка обеспечивает синхронизацию и подавление всплесков в SCL (ic_clk_in_a). Следовательно, когда IC_SDA_TX_HOLD имеет значение меньше, чем SPKLEN + 7, I^2C управляет SDA (ic_data_oe) SPKLEN + 7 циклами ic_clk после того, как SCL (ic_clk_in) перешел в логический 0.

На Рисунок 32.31 показано время THD:DAT, генерируемое I^2C, когда IC_SDA_TX_HOLD = 3.

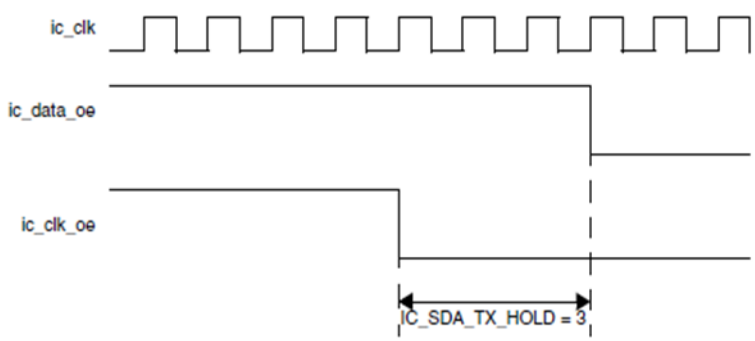


Рисунок 32.31. IC_SDA_TX_HOLD = 3

32.16 Регистры

Регистры в I^2C находятся в pclk, но биты состояния отражают действия, которые происходят в ic_clk. Следовательно, когда pclk отражает активность, произошедшую на ic_clk.

Некоторые регистры могут быть записаны только тогда, когда I^2C отключен, запрограммирован регистром IC_ENABLE. Программное обеспечение не должно отключать I^2C, пока оно активно. Если I^2C находится в процессе передачи, когда он отключен, то I^2C останавливается, а также удаляет содержимое буфера передачи после завершения текущей передачи. Слейв продолжает прием до тех пор, пока удаленный мастер не прервет передачу, и в этом случае I^2C может быть отключен.

Если pclk и ic_clk не идентичны (IC_CLK_TYPE = 0), для синхронного и асинхронного режимов существует задержка в два регистра.

32.16.1 Регистры прерывания

В Таблица 32.64 перечислены операции регистров прерывания I^2C, как они устанавливаются и очищаются.

Таблица 32.64. Регистры прерывания

Interrupt Bit Fields	Set by Hardware / Cleared by Software	Set and Cleared by Hardware
MST_ON_HOLD	–	+
RESTART_DET	+	–
GEN_CALL	+	–
START_DEL	+	–
STOP_DEL	+	–
ACTIVITY	+	–
RX_DONE	+	–
TX_ABRT	+	–
RD_REQ	+	–
TX_EMPTY	–	+
TX_OVER	+	–
RX_FULL	–	+
RX_OVER	+	–
RX_UNDER	+	–

На Рисунок 32.32 показана работа регистров прерывания, где биты устанавливаются аппаратно и очищаются программным обеспечением.

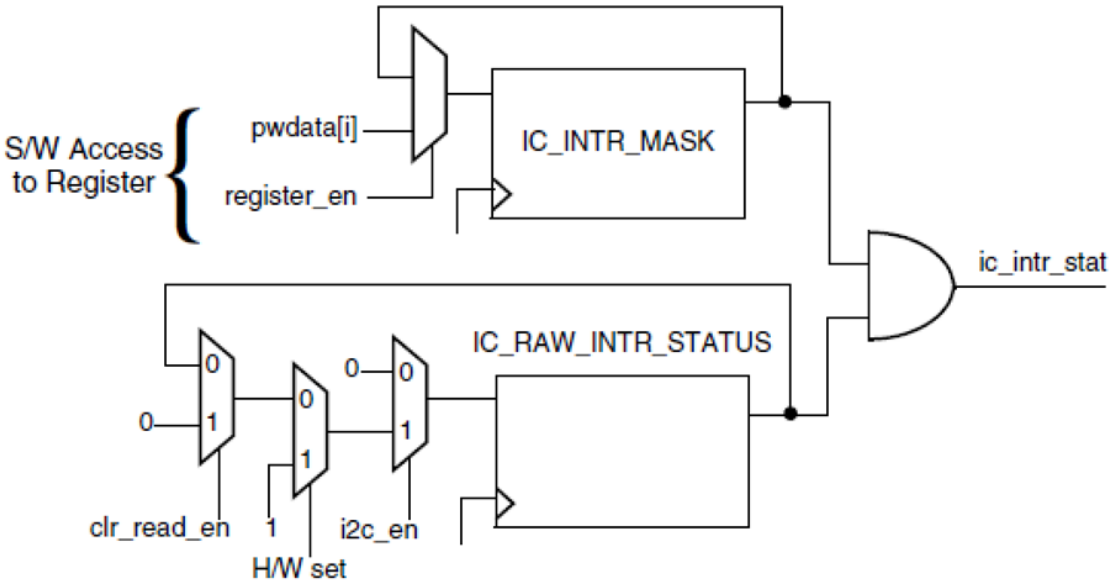


Рисунок 32.32. Схема прерывания

33. КОНТРОЛЛЕР SSI (SSI)

33.1 Общая информация

В микросхеме реализованы три контроллера SSI:

- SSI0 в подсистеме lsperiph0 – используется для подключения внешних устройств
- SSI1 в подсистеме lsperiph1 – также используется для подключения внешних устройств и обладает аналогичными возможностями, что и SSI0
- SPI_OTP в сервисной подсистеме – используется только для доступа в OTP память

Характеристики контроллеров SSI представлены в Таблица 33.1:

Таблица 33.1 Характеристики контроллеров SSI

Характеристика	SSI0, SSI1	SPI_OTP
Поддерживаемый режим работы	Мастер	Мастер
Поддерживаемый стандарт	Standard SPI	Standard / Dual / Quad / Octal SPI
Поддерживаемые протоколы	Motorola SPI, Texas Instruments Synchronous Serial, National Semiconductor Microwire.	Motorola SPI, Texas Instruments Synchronous Serial, National Semiconductor Microwire.
Динамическое управление скоростью передачи данных	Есть	Есть
Размер одного слова	от 4 до 32 бит	от 4 до 16 бит
FIFO буфер приемника и передатчика	64 слова	8 слов
Количество каналов slave-select	4	2
Аппаратный интерфейс запросов к DMA	Есть	Нет
Поддержка DDR режима	Нет	Есть

33.2 Регистры

Перечень регистров приведен в Таблица 33.2.

Таблица 33.2. Перечень программно-доступных регистров

Смещение	Условное обозначение	Название регистра	Исходное состояние	Тип доступа
0x0	CTRLR0	Управляющий регистр 0	0x01070000	RW
0x4	CTRLR1	Управляющий регистр 1	0x00000000	RW
0x8	SSIENR	Разрешающий регистр	0x00000000	RW
0xc	MWCR	Управление протоколом Microwire	0x00000000	RW
0x10	SER	Управление слэйвом	0x00000000	RW
0x14	BAUDR	Выбор скоростного режима	0x00000000	RW
0x18	TXFTLR	Порог заполнения, FIFO передатчика	0x00000000	RW
0x1c	RXFTLR	Порог заполнения, FIFO приемника	0x00000000	RW

Смещение	Условное обозначение	Название регистра	Исходное состояние	Тип доступа
0x20	TXFLR	Уровень заполнения FIFO передатчика	0x00000000	RO
0x24	RXFLR	Уровень заполнения FIFO приемника	0x00000000	RO
0x28	SR	Регистр статуса	0x00000006	RO
0x2c	IMR	Регистр маски прерываний	0x0000003F	RW
0x30	ISR	Регистр статуса прерываний	0x00000000	RO
0x34	RISR	Регистр прерываний	0x00000000	RO
0x38	TXOICR	Регистр снятия прерывания переполнения FIFO передатчика	0x00000000	RO
0x3c	RXOICR	Регистр снятия прерывания переполнения FIFO приемника	0x00000000	RO
0x40	RXUICR	Регистр снятия прерывания опустошения FIFO приемника	0x00000000	RO
0x44	MSTICR	Регистр снятия прерывания Multi-Master	0x00000000	RO
0x48	ICR	Регистр снятия прерываний	0x00000000	RO
0x4c	DMACR*	Регистр управление DMA	0x00000000	RW
0x50	DMATDLR*	Уровень передачи DMA	0x00000000	RW
0x54	DMARDLR*	Уровень приема DMA	0x00000000	RW
0x58	IDR	Регистр идентификации	0xFFFFFFFF	RO
0x5c	SSI_VERSION_ID	Версия компонента	0x3430312A	RO
0x60	DR0	Регистр данных x	0x00000000	RW
0x64	DR1	Регистр данных x	0x00000000	RW
0x68	DR2	Регистр данных x	0x00000000	RW
0x6c	DR3	Регистр данных x	0x00000000	RW
0x70	DR4	Регистр данных x	0x00000000	RW
0x74	DR5	Регистр данных x	0x00000000	RW
0x78	DR6	Регистр данных x	0x00000000	RW
0x7c	DR7	Регистр данных x	0x00000000	RW
0x80	DR8	Регистр данных x	0x00000000	RW
0x84	DR9	Регистр данных x	0x00000000	RW
0x88	DR10	Регистр данных x	0x00000000	RW
0x8c	DR11	Регистр данных x	0x00000000	RW
0x90	DR12	Регистр данных x	0x00000000	RW
0x94	DR13	Регистр данных x	0x00000000	RW
0x98	DR14	Регистр данных x	0x00000000	RW
0x9c	DR15	Регистр данных x	0x00000000	RW
0xa0	DR16	Регистр данных x	0x00000000	RW
0xa4	DR17	Регистр данных x	0x00000000	RW
0xa8	DR18	Регистр данных x	0x00000000	RW
0xac	DR19	Регистр данных x	0x00000000	RW
0xb0	DR20	Регистр данных x	0x00000000	RW
0xb4	DR21	Регистр данных x	0x00000000	RW
0xb8	DR22	Регистр данных x	0x00000000	RW
0xbc	DR23	Регистр данных x	0x00000000	RW
0xc0	DR24	Регистр данных x	0x00000000	RW
0xc4	DR25	Регистр данных x	0x00000000	RW
0xc8	DR26	Регистр данных x	0x00000000	RW
0xcc	DR27	Регистр данных x	0x00000000	RW
0xd0	DR28	Регистр данных x	0x00000000	RW
0xd4	DR29	Регистр данных x	0x00000000	RW
0xd8	DR30	Регистр данных x	0x00000000	RW
0xdc	DR31	Регистр данных x	0x00000000	RW

Смещение	Условное обозначение	Название регистра	Исходное состояние	Тип доступа
0xe0	DR32	Регистр данных x	0x00000000	RW
0xe4	DR33	Регистр данных x	0x00000000	RW
0xe8	DR34	Регистр данных x	0x00000000	RW
0xec	DR35	Регистр данных x	0x00000000	RW
0xf0	RSVD	Резерв	0x00000000	RO
0xf4	SPI_CTRL0**	Регистр управления режимом работы	0x00000200	RW
0xf8	TXD_DRIVE_EDGE**	Регистр управления активным фронтом в DDR режиме	0x00000000	RW
0xfc	RSVD	Резерв	0x00000000	RO

* - регистры присутствуют только в SSI0 и SSI1

** - регистры присутствуют только в SPI_OTP

33.2.1 Регистр CTRLR0

Регистр управления последовательной передачей данных. Запись в регистр невозможна, если регистр SSIENR находится в активном состоянии.

Формат регистра CTRLR0 приведен в следующей таблице:

Таблица 33.3. Формат регистра CTRLR0

Разряды	Название	Описание	Сброс
31:25	RSVD_CTRLR0	Резерв – только чтение	0x0
24	SSTE	Slave Select Toggle Enable Управление переключением slave select линии. Во время функционирования в режиме SSI с тактовой частотой (SCPH) установленной в 0, этот регистр управляет поведением линии slave select (ss_*_n) между блоками данных. Если это поле установлено в 1, линия ss_*_n будет переключаться между последовательными блоками данных и sclk будет удерживаться в состоянии по умолчанию. Если поле установлено в 0 ss_*_n будет оставаться в низком положении, а sclk будет продолжать работу на протяжении всей передачи данных.	0x1
23	RSVD_CTRLR0_23	CTRLR0_23 резерв – только чтение	0x0
22:21	SSI_FRF	SSI Frame Format Резерв	0x0
20:16	DFS_32	Data Frame Size Размер блока данных в 32-х битном режиме передачи данных.	0x7
15:12	CFS	Control Frame Size. Выбор длинны управляющего слова для протокола Microwire.	0x0
11	SRL	Shift Register Loop. Используется только в целях тестирования. В активном состоянии соединяет передающий сдвиговый регистр со сдвиговым регистром приемника.	0x0
10	RSVD_SLV_OE	SLV_OE Резерв- только чтение	0x0

Разряды	Название	Описание	Сброс
9:8	TMOD	Transfer Mode. Выбор режима последовательной передачи данных. Это поле не влияет на удвоение передачи, а только определяет валидность принимаемых или передаваемых данных. 00 – Передача и прием 01 – Только передача 10 – Только прием 11 – EEPROM чтение В режиме “Только передача”, принимаемые данные не валидны и не поступают в FIFO приемника перезаписываются после следующей передачи данных. В режиме “Только прием” передаваемые данные не валидны. После первой передачи, это же слово будет отправлено снова за стандартное время транзакции. В режиме “Прием и передача”, передаваемые и принимаемые данные валидны. Передача продолжится до последнего слова в FIFO передатчика. Принимаемые данные остаются в FIFO приемника до тех пор, пока не будут прочитаны. В режиме “EPROM чтение”, принимаемые данные не будут валидны до окончания передачи управляющих инструкций. Когда все инструкции были отправлены, принимаемые данные становятся валидными, передаваемые данные становятся не валидными. Все данные передатчика предполагаются управляющими инструкциями.	0x0
7	SCPOL	Serial Clock Polarity. Выбор полярности неактивного тактового сигнала, который становится неактивным, в случае отсутствия передаваемых данных в режиме master Используется, только если для формата блока данных (FRF) выбран to Motorola SPI.	0x0
6	SCPH	Serial Clock Phase. Фаза тактового сигнала, задает отношение тактового сигнала и slave select. 0x0 – захват данных происходит по первому фронту тактового сигнала 0x1 – происходит пропуск одного периода тактового сигнала после установки slave select, захват данных происходит по второму фронту тактового сигнала	0x0
5:4	FRF	Frame Format. Выбор протокола передачи данных	0x0

Разряды	Название	Описание	Сброс
		0x0 – MOTOROLA_SPI 0x1 – TEXAS_SSP 0x2 – NS_MICROWIRE 0x3 – Резерв	
3:0	DFS	Data Frame Size Резерв - только чтение	0x0

33.2.2 Регистр CTRLR1

Регистр управления завершением последовательной передачи. Запись в регистр невозможна, если регистр SSIENR находится в активном состоянии.

Формат регистра CTRLR1 приведен в следующей таблице:

Таблица 33.4. Формат регистра CTRLR1

Разряды	Название	Описание	Сброс
31:16	RSVD_CTRLR1	CTRLR1 Резерв – только чтение	0x0
15:0	NDF	Number of Data Frames. Выбор количество блоков данных для приема, когда TMOD = 10 или TMOD = 11. Количество блоков данных, принятых по последовательному интерфейсу равно NDF + 1 (Максимум 64 Кбит)	0x0

33.2.3 Регистр SSIENR

Регистр включения/отключения SSI.

Формат регистра SSIENR приведен в следующей таблице:

Таблица 33.5. Формат регистра SSIENR

Разряды	Название	Описание	Сброс
31:1	RSVD_SSIENR	SSIENR Резерв – только чтение.	0x0
0	SSI_EN	SSI Enable. Включает и отключает все операции SSI. При отключении любые операции передачи данных прерываются немедленно. FIFO буферы приемника и передатчика очищаются. В активном состоянии, блокируется возможность записи в любые регистры управления.	0x0

33.2.4 Регистр MWCR

Регистр управления направлением передачи данных в режиме полудуплекс Microwire. Запись в регистр невозможна, если регистр SSIENR находится в активном состоянии.

Формат регистра MWCR приведен в следующей таблице:

Таблица 33.6. Формат регистра MWCR

Разряды	Название	Описание	Сброс
31:3	RSVD_MWCR	MWCR Резерв – только чтение.	0x0
2	MHS	Microwire Handshaking. Включает/отключает busy/ready проверку, протокола Microwire. В активном состоянии SSI проверяет готовность целевого slave устройства, после передачи последнего бита данных, перед снятием BUSY статуса в регистре SR.	0x0
1	MDD	Microwire Control. Определяет направление передачи слова данных в режиме Microwire. 0x0 – SSI передает слово данных 0x1 – SSI принимает слово данных	0x0
0	MWMOD	Microwire Transfer Mode. Определяет тип передачи Microwire. 0x0 – одиночная Microwire передача 0x1 – последовательная передача Microwire	0x0

33.2.5 Регистр SER

Регистр управление slave-select сигналом. Регистр может быть установлен в 1 или 0, если SSI_EN=0. Если SSI_EN=1, регистр может быть установлен только в 1 (для задержки установки slave-select сигнала во время заполнения FIFO передатчика).

Формат регистра SER приведен в следующей таблице:

Таблица 33.7. Формат регистра SER

Разряды	Название	Описание	Сброс
31:4	RSVD_SER	SER Резерв – только чтение.	0x0
3:0	SER	Slave Select Enable Flag. Каждый бит этого поля отвечает за соответствующий сигнал slave-select(ss_x_n). Нужно отметить,	0x0

Разряды	Название	Описание	Сброс
		что очищение любого бита в этом во время передачи данных не приведет к немедленному сбросу соответствующего slave select сигнала, запись в это поле должна производиться перед началом передачи данных. Если не выбран режим вещания, то только один бит этого поле должен быть установлен.	

33.2.6 Регистр BAUDR

Регистр определения частоты тактового сигнала последовательного интерфейса. Значение в этом регистре является делителем частоты SSI. Запись в регистр невозможна, если регистр SSIENR находится в активном состоянии.

Формат регистра BAUDR приведен в следующей таблице:

Таблица 33.8. Формат регистра BAUDR

Разряды	Название	Описание	Сброс
31:16	RSVD_BAUDR	BAUDR Резерв – Только чтение	0x0
15:0	SCKDV	SSI Clock Divider. Младший бит этого регистра всегда равен 0 и доступен только для чтения – это гарантирует четность значения в этом регистре. Если значение SCKDV = 0, то тактовый сигнал SSI будет отключен. Частота тактового сигнала SSI определяется формулой: Fsclk_out = Fssi_clk/SCKDV Где SCDV – любое четное число от 2 до 65534	0x0

33.2.7 Регистр TXFTLR

Регистр управления порогом заполнения FIFO буфера передатчика.

Формат регистра TXFTLR приведен в следующей таблице:

Таблица 33.9. Формат регистра TXFTLR

Разряды	Название	Описание	Сброс
31:6	RSVD_TXFTLR	TXFTLR Резерв – только чтение.	0x0

Разряды	Название	Описание	Сброс
5:0	TFT	Transmit FIFO Threshold. Задает количество слов данных на котором FIFO буфер передатчика сгенерирует прерывание. При попытке записи значения, превышающего глубину FIFO – значение регистра останется неизменным. Когда количество слов данных в FIFO меньше или равно значению в этом поле, генерируется прерывание (ssi_txe_intr).	0x0

33.2.8 Регистр RXFTLR

Регистр управления порогом заполнения FIFO буфера приемника.

Reset Value: 0x0

Формат регистра RXFTLR приведен в следующей таблице:

Таблица 33.10. Формат регистра RXFTLR

Разряды	Название	Описание	Сброс
31:6	RSVD_RXFTLR	RXFTLR Резерв – только чтение.	0x0
5:0	RFT	Receive FIFO Threshold. Задает количество слов данных на котором FIFO буфер приемника сгенерирует прерывание. При попытке записи значения, превышающего глубину FIFO – значение регистра останется неизменным. Когда количество слов данных в FIFO больше или равно значению в этом поле, генерируется прерывание (ssi_rxe_intr).	0x0

33.2.9 Регистр TXFLR

Регистр количества слов данных в FIFO передатчика.

Формат регистра TXFLR приведен в следующей таблице:

Таблица 33.11. Формат регистра TXFLR

Разряды	Название	Описание	Сброс
31:7	RSVD_TXFLR	TXFLR Резерв – только чтение.	0x0

Разряды	Название	Описание	Сброс
6:0	TXTFL	Transmit FIFO Level. Количество слов данных в FIFO буфере передатчика.	0x0

33.2.10 Регистр RXFLR

Регистр количества слов данных в FIFO приемника.

Формат регистра RXFLR приведен в следующей таблице:

Таблица 33.12. Формат регистра RXFLR

Разряды	Название	Описание	Сброс
31:7	RSVD_RXFLR	RXFLR Reserved bits - Read Only	0x0
6:0	RXTFL	Receive FIFO Level. Количество слов данных в FIFO буфере приемника.	0x0

33.2.11 Регистр SR

Регистр индикации статуса передачи данных, статуса FIFO и любых ошибок приема/передачи которые могут возникнуть.

Формат регистра SR приведен в следующей таблице:

Таблица 33.13. Формат регистра SR

Разряды	Название	Описание	Сброс
31:7	RSVD_SR	SR Резерв – только чтение.	0x0
6	DCOL	Data Collision Error. Бит будет установлен в 1, только если, сигнал ss_in_n будет установлен другим мастером, во время передачи данных блоком SSI. Ошибка является указателем того, что последняя передача данных была прервана до завершения. Бит сбрасывается после чтения.	0x0
5	RSVD_TXE	TXE Резерв – только чтение	0x0
4	RFF	Receive FIFO Full. FIFO приемника полон, бит сбрасывается, если в буфере есть свободные места.	0x0
3	RFNE	Receive FIFO Not Empty. FIFO приемника не пуст, бит устанавливается если в буфере	0x0

Разряды	Название	Описание	Сброс
		есть хотя бы одно слово данных. Сбрасывается если буфер пуст.	
2	TFE	Transmit FIFO Empty. FIFO буфер передатчика пуст, сбрасывается если в буфере есть хотя бы одно слово данных.	0x1
1	TFNF	Transmit FIFO Not Full. FIFO передатчика не полон, бит сбрасывается, если буфер полон.	0x1
0	BUSY	SSI Busy Flag. 0x1 – идет передача SSI 0x0 – SSI в состоянии ожидания или выключен	0x0

33.2.12 Регистр IMR

Регистр маскирования прерываний.

Формат регистра IMR приведен в следующей таблице:

Таблица 33.14. Формат регистра IMR

Разряды	Название	Описание	Сброс
31:6	RSVD_IMR	IMR Резерв – только чтение.	0x0
5	MSTIM	Multi-Master Contention Interrupt	0x1
4	RXFIM	Receive FIFO Full Interrupt Mask	0x1
3	RXOIM	Receive FIFO Overflow Interrupt Mask	0x1
2	RXUIM	Receive FIFO Underflow Interrupt Mask	0x1
1	TXOIM	Transmit FIFO Overflow Interrupt Mask	0x1
0	TXEIM	Transmit FIFO Empty Interrupt Mask	0x1

33.2.13 Регистр ISR

Регистр статуса прерываний после маскирования.

Формат регистра ISR приведен в следующей таблице:

Таблица 33.15. Формат регистра ISR

Разряды	Название	Описание	Сброс
31:6	RSVD_ISR	ISR Reserved bits - Read Only	0x0
5	MSTIS	Multi-Master Contention Interrupt Status.	0x0
4	RXFIS	Receive FIFO Full Interrupt Status	0x0
3	RXOIS	Receive FIFO Overflow Interrupt Status	0x0

Разряды	Название	Описание	Сброс
2	RXUIS	Receive FIFO Underflow Interrupt Status	0x0
1	TXOIS	Transmit FIFO Overflow Interrupt Status	0x0
0	TXEIS	Transmit FIFO Empty Interrupt Status	0x0

33.2.14 Регистр RISR

Регистр статуса прерываний до маскирования.
Формат регистра RISR приведен в следующей таблице:

Таблица 33.16. Формат регистра RISR

Разряды	Название	Описание	Сброс
31:6	RSVD_RISR	RISR Reserved bits - Read Only	0x0
5	MSTIR	Multi-Master Contention Raw Interrupt Status.	0x0
4	RXFIR	Receive FIFO Full Raw Interrupt Status	0x0
3	RXOIR	Receive FIFO Overflow Raw Interrupt Status	0x0
2	RXUIR	Receive FIFO Underflow Raw Interrupt Status	0x0
1	TXOIR	Transmit FIFO Overflow Raw Interrupt Status	0x0
0	TXEIR	Transmit FIFO Empty Raw Interrupt Status	0x0

33.2.15 Регистр TXOICR

Регистр сброса прерывания: “Переполнен FIFO передатчика”.

Формат регистра TXOICR приведен в следующей таблице:

Таблица 33.17. Формат регистра TXOICR

Разряды	Название	Описание	Сброс
31:1	RSVD_TXOICR	TXOICR Резерв – только чтение.	0x0
0	TXOICR	Clear Transmit FIFO Overflow Interrupt. Значение поля отображает статус прерывания, Чтение регистра сбросит прерывание ssi_txo_intr. Запись не имеет эффекта.	0x0

33.2.16 Регистр RXOICR

Регистр сброса прерывания: “Переполнен FIFO приемника”.

Формат регистра RXOICR приведен в следующей таблице:

Таблица 33.18. Формат регистра RXOICR

Разряды	Название	Описание	Сброс
31:1	RSVD_RXOICR	RXOICR Резерв – только чтение.	0x0
0	RXOICR	Clear Receive FIFO Overflow Interrupt. Значение поля отображает статус прерывания, Чтение регистра сбросит прерывание ssi_rxo_intr. Запись не имеет эффекта.	0x0

33.2.17 Регистр RXUICR

Регистр сброса прерывания: “FIFO приемника пуст”

Формат регистра RXUICR приведен в следующей таблице:

Таблица 33.19. Формат регистра RXUICR

Разряды	Название	Описание	Сброс
31:1	RSVD_RXUICR	RXUICR Резерв – только чтение.	0x0
0	RXUICR	Clear Receive FIFO Underflow Interrupt. Значение поля отображает статус прерывания, Чтение регистра сбросит прерывание ssi_rxu_intr. Запись не имеет эффекта.	0x0

33.2.18 Регистр MSTICR

Регистр сброса прерывания: “Возможный конфликт master устройств”

Формат регистра MSTICR приведен в следующей таблице:

Таблица 33.20. Формат регистра MSTICR

Разряды	Название	Описание	Сброс
31:1	RSVD_MSTICR	MSTICR Резерв – только чтение.	0x0
0	MSTICR	Clear Multi-Master Contention Interrupt. Значение поля отображает статус прерывания, Чтение регистра сбросит прерывание ssi_mst_intr. Запись не имеет эффекта.	0x0

33.2.19 Регистр ICR

Регистр сброса прерываний.

Формат регистра ICR приведен в следующей таблице:

Таблица 33.21. Формат регистра ICR

Разряды	Название	Описание	Сброс
31:1	RSVD_ICR	ICR Резерв – только чтение.	0x0
0	ICR	Clear Interrupts. Значение поля отображает наличие любого из следующих прерываний: ssi_txo_intr ssi_rxu_intr ssi_rxo_intr ssi_mst_intr Чтение сбрасывает все вышеперечисленные прерывание, запись не имеет эффекта.	0x0

33.2.20 Регистр DMACR

Регистр управления аппаратным интерфейсом DMA.

Формат регистра DMACR приведен в следующей таблице:

Таблица 33.22. Формат регистра DMACR

Разряды	Название	Описание	Сброс
31:2	RSVD_DMACR	DMACR Резерв только чтение.	0x0
1	TDMAE	Transmit DMA Enable. Бит включения/отключения DMA канала FIFO буфера передатчика.	0x0
0	RDMAE	Receive DMA Enable. Бит включения/отключения DMA канала FIFO буфера приемника.	0x0

33.2.21 Регистр DMATDLR

Регистр управления запросом к DMA по уровню заполнения FIFO буфера передатчика.

Формат регистра DMATDLR приведен в следующей таблице:

Таблица 33.23. Формат регистра DMATDLR

Разряды	Название	Описание	Сброс
31:6	RSVD_DMATDLR	DMATDLR Резерв – только чтение.	0x0
5:0	DMATDL	Transmit Data Level. Поле управления запросом DMA (dma_tx_req), по уровню заполнения FIFO передатчика. Запрос формируется если количество слов данных в буфере передатчика меньше или равно DMATDL и TDMAE = 1.	0x0

33.2.22 Регистр DMARDLR

Регистр управления запросом к DMA по уровню заполнения FIFO буфера приемника.

Формат регистра DMARDLR приведен в следующей таблице:

Таблица 33.24. Формат регистра DMARDLR

Разряды	Название	Описание	Сброс
31:6	RSVD_DMARDLR	DMARDLR Резерв – только чтение.	0x0
5:0	DMARDL	Receive Data Level. Поле управления запросом DMA (dma_rx_req), по уровню заполнения FIFO приемника. Запрос формируется если количество слов данных в буфере приемника больше или равно DMARDL + 1 и RDMAE = 1.	0x0

33.2.23 Регистр IDR

Идентификационный регистр

Формат регистра IDR приведен в следующей таблице:

Таблица 33.25. Формат регистра IDR

Разряды	Название	Описание	Сброс
31:0	IDCODE	Identification code.	0xffffffff

33.2.24 Регистр SSI_VERSION_ID

Регистр версии SSI

Формат регистра SSI_VERSION_ID приведен в следующей таблице:

Таблица 33.26. Формат регистра SSI_VERSION_ID

Разряды	Название	Описание	Сброс
31:0	SSI_COMP_VERSION	HEX представление, версии SSI. Состоит из ASCII значения для каждого числа.	0x3430312a

33.2.25 Регистр DR0 – DR35

Регистр данных SSI.

Чтение из регистра считывает данные из FIFO буфера приемника. Записанные в регистр данные помещаются в FIFO буфер передатчика.

Запись возможно только, если SSI_EN=1.

Регистр сбрасывается при SSI_EN = 0.

Регистр резервирует 36 адресов, для поддержки BURST транзакций, запись в любой регистр равноценно помещению слов данных в FIFO передатчика, чтение любого из них равноценно чтению данных из FIFO приемника. Данные FIFO буферов не адресуются.

Формат регистра DR приведен в следующей таблице:

Таблица 33.27. Формат регистра DR

Разряды	Название	Описание	Сброс
31:16	DR[31:16]	RSVD Резерв – для SPI_OTP Data Register [31:16] – для SSI0 и SSI1	0x0
15:0	DR[15:0]	Data Register [15:0]	0x0

33.2.26 Регистр SPI_CTRL0

Регистр используется для управления последовательной передачей данных только в режимах Dual/Quad/Octal работы SPI. Записать в этот регистр возможна только когда SSI_EN=0. Блок включается и выключается путем записи в регистр SSIENR. Формат регистра SPI_CTRL0 приведен в следующей таблице:

Таблица 33.28. Формат регистра SPI_CTRL0

Разряды	Название	Описание	Сброс
31:18	RSVD	RSVD Резерв	0x0
17	INST_DDR_EN	Включает передачу данных с двойной скоростью для фазы инструкции.	0x0
16	SPI_DDR_EN	Включает передачу данных с двойной скоростью в форматах кадров Dual/Quad/Octal SPI.	0x0

Разряды	Название	Описание	Сброс
15:11	WAIT_CYCLES	Количество тактов ожидания в режиме Dual/Quad/Octal между фазой передачи кадров управления и фазой приемом данных. Значение указывается в количестве тактов SPI.	0x0
10	RSVD	RSVD Резерв	0x0
9:8	INST_L	Длина инструкции в режимах Dual/Quad/Octal 0x0 – 0 бит (нет инструкции) 0x1 – 4-битная инструкция 0x2 – 8-битная инструкция 0x3 – 16-битная инструкция	0x2
7:6	RSVD	RSVD Резерв	0x0
5:2	ADDR_L	Длину адреса для передачи. Передача может начаться только после программирования данного поля. Возможные значения: 0x0: 0-bit Address Width 0x1: 4-bit Address Width 0x2: 8-bit Address Width 0x3: 12-bit Address Width 0x4: 16-bit Address Width 0x5: 20-bit Address Width 0x6: 24-bit Address Width 0x7: 28-bit Address Width 0x8: 32-bit Address Width 0x9: 36-bit Address Width 0xa: 40-bit Address Width 0xb: 44-bit Address Width 0xc: 48-bit Address Width 0xd: 52-bit Address Width 0xe: 56-bit Address Width 0xf: 60-bit Address Width	0x0
1:0	TRANS_TYPE		0x0

33.2.27 Регистр TXD_DRIVE_EDGE

Регистр управления активным фронтом в DDR режиме. Регистр используется для управления активным фронтом защелкивания TXD в DDR режиме. Записать в этот регистр возможна только когда SSI_EN=0. Блок включается и выключается путем записи в регистр SSIENR

Формат регистра TXD_DRIVE_EDGE приведен в следующей таблице:

Таблица 33.29. Формат регистра TXD_DRIVE_EDGE

Разряды	Название	Описание	Сброс
31:8	RSVD	RSVD Резерв	0x0

Разряды	Название	Описание	Сброс
7:0	TDE	TXD Drive edge - значение, которое определяет ведущий фронт передаваемых данных. Максимальное значение этого регистра = (BAUDR/2) -1.	0x0

33.2.28 Регистр RSVD

Резерв

Формат регистра RSVD приведен в следующей таблице:

Таблица 33.30. Формат регистра RSVD

Разряды	Название	Описание	Сброс
31:0	RSVD	RSVD Резерв	0x0

33.3 Функционирование

33.3.1 Режим обмена

Выбор режима обмена происходит посредством битов TMOD регистра CTRLR0. В зависимости от значения этих битов, возможны следующие режимы обмена:

Прием и передача:

Когда TMOD = 0x0, возможны прием и передача данных. Обмен данными происходит в соответствии с выбранным протоколом (поле FRF регистра CTRL0) данные поступают из FIFO передатчика. От slave-устройства данные поступают в FIFO приемника.

Только передача:

Когда TMOD = 0x1, принимаемые от slave-устройства данные не поступают в FIFO приемника. Передача данных происходит в обычном режиме по выбранному последовательному протоколу. При использовании этого режима необходимо маскировать прерывания от приемника.

Только прием:

Когда TMOD = 0x2, данные не записываются в FIFO передатчика. Прием данных происходит в обычном режиме по выбранному протоколу. При использовании этого режима необходимо маскировать прерывания от передатчика.

Чтение EEPROM:

Когда TMOD = 0x3, передача данных используется для передачи кода операции и/или адреса EEPROM-устройства. Обычно это занимает 3 кадра данных (8 бит – старший адрес, 8 бит – младший адрес, 8 бит – код операции). Во время передачи кода операции и адреса не происходит приема данных (игнорируется данные с линии RXD). Порт продолжает передачу данных до тех пор, пока FIFO передатчика не станет пустым. Поэтому перед началом передачи FIFO передатчика должно иметь только необходимые данные (адрес и код операции), иначе происходит потеря данных, читаемых из EEPROM-устройства. После того как FIFO передатчика станет пустым (будет отправлена вся контрольная информация), станет доступен прием данных с линии RXD. Последовательный обмен будет продолжаться до тех пор, пока число принятых слов данных не превысит на 1 значение бит NDF регистра CTRLR1. Режим недоступен для SSP протокола.

Замечание:

Не рекомендуется смешивать ведомые разных протоколов на одном контроллере. Для работы SSI, контактная площадка сигнала SSIx_SS_IN должна быть в аппаратной функции.

33.3.2 Тактирование SSI контроллера

Мксимальная частота интерфейса SSI (sclk_out) определяется как половина частоты контроллера SSI (ssi_clk)

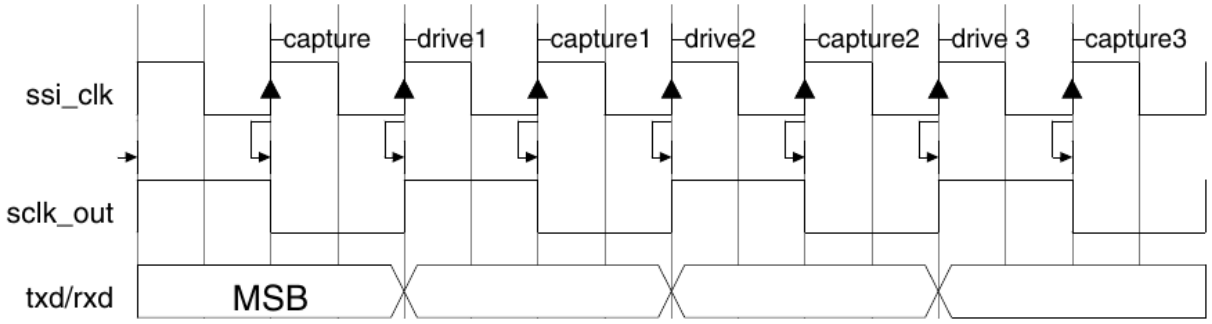


Рисунок 33.1 Максимальное отношение частот sclk_out/ssi_clk

В общем виде частота интерфейса определяется как $F_{sclk\_out} = \frac{F_{ssi\_clk}}{SCKDV}$

SCKDV – поле регистра BAUDR, может принимать значение от 0 до 65534, если значение поля 0 то выходная частота отсутствует.

33.3.3 Прерывания SSI

SSI оснащен комбинированным сигналом прерывания с поддержкой маскирования, прерывание возникает в одном из следующих случаев:

- Transmit FIFO empty Interrupt (spi_txe_intr) – уровень FIFO передатчика меньше или равен установленному значению (поле TFT регистра TXFTLR)

- Transmit FIFO overflow Interrupt (ssi_txo_int) – устанавливается в случае переполнения FIFO передатчика. Снимается чтением TXOICR регистра.
- Receive FIFO Full Interrupt (ssi_rxf_intr) – уровень FIFO приемника больше или равен RXT + 1 (регистра RXFLTR), снимается по уменьшению уровня в FIFO приемника.
- Receive FIFO Overflow Interrupt (ssi_rxo_intr) – переполнение FIFO приемника, приходящие данные не будут записаны, снимается по чтению RXOICR регистра
- Receive FIFO Underflow Interrupt (ssi_rxu_intr) – устанавливается при чтении из пустого FIFO приемника, снимается чтением RXUICR регистра
- Multi-Master Contention Interrupt (ssi_mst_intr) – устанавливается в случае обращения другого мастер устройства к контроллеру SSI как к ведомому устройству, что может сигнализировать о конфликтах на SSI шине. Снимается чтением MSTICR регистра.

33.3.4 Motorola Serial Peripheral Interface (SPI)

Протокол SPI имеет два определяющих параметра:

SCPOL – исходный уровень сигнала синхронизации. Если SCPOL=0, то линия синхронизации до начала цикла передачи и после его окончания имеет низкий уровень иначе, если SCPOL=1, – высокий.

SCPH – фаза синхронизации. От этого параметра зависит, в какой последовательности выполняется установка и выборка данных (если SCPH=0, то по переднему фронту в цикле синхронизации будет выполняться выборка данных, а затем, по заднему фронту, – установка данных; если же SCPH=1, то установка данных будет выполняться по переднему фронту в цикле синхронизации, а выборка - по заднему).

Для всех случаев необходимо установить напряжение ножки (set pull up) для сигнала SSIx_SS_IN в значение 1. Не рекомендуется смешивать ведомые разных протоколов на одном контроллере.

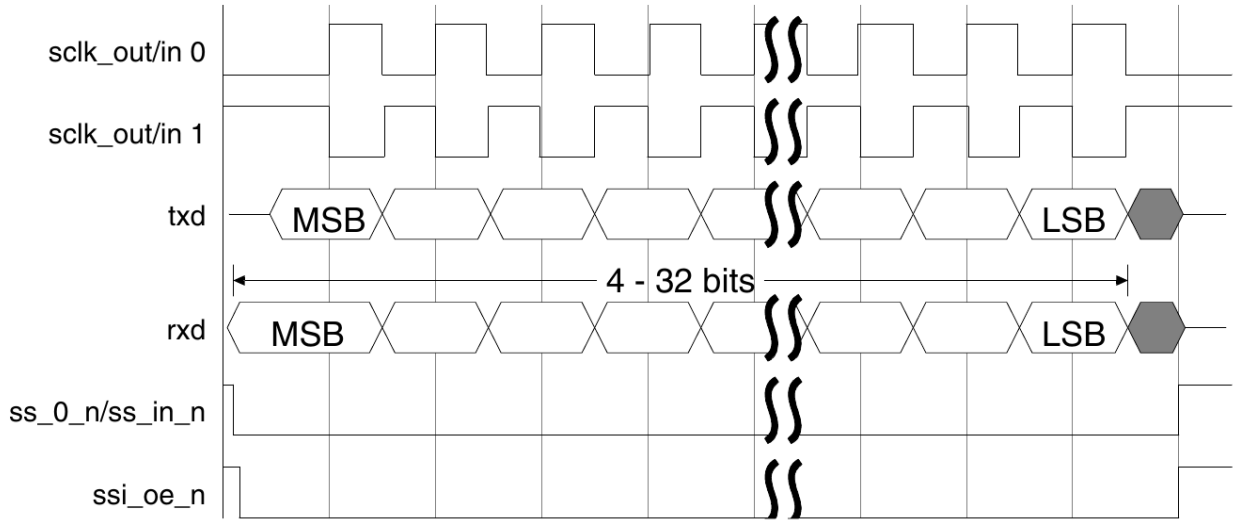


Рисунок 33.2 Пример последовательного обмена по протоколу SPI с параметром SCPH = 0

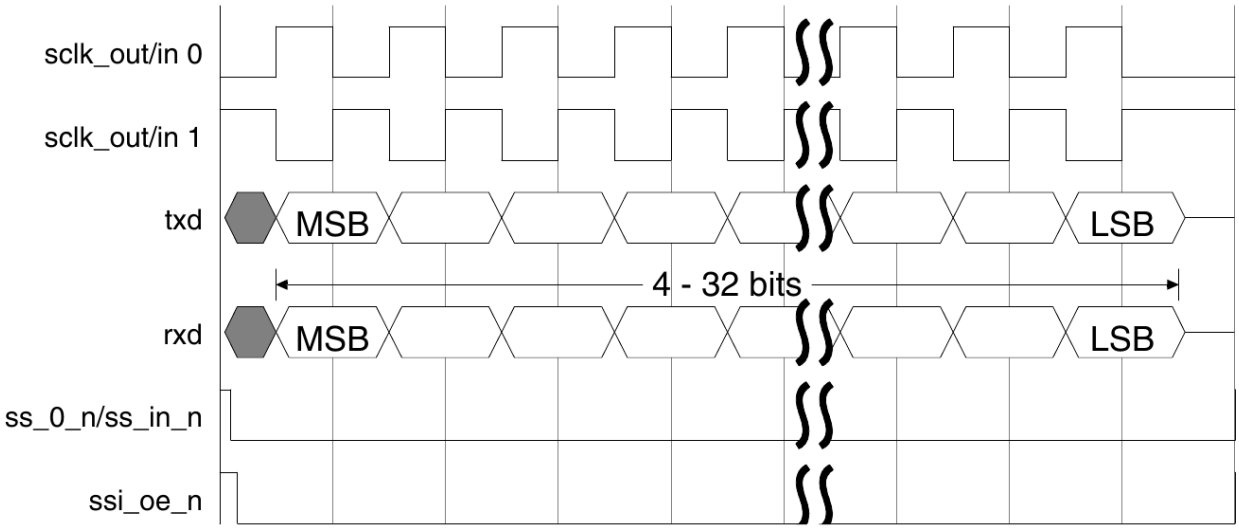


Рисунок 33.3 Пример последовательного обмена по протоколу SPI с параметром SCPH = 1

33.3.5 Texas Instruments Synchronous Serial Protocol (SSP)

Обмен по протоколу SSP начинается с установки в 1 индикатора линии SSIx_SS_IN на 1 такт. Передача/прием данных начинается на следующий такт, после перехода сигнала SSIx_SS_IN в 0. Установка данных всегда происходит по переднему фронту сигнала синхронизации, а выборка – по заднему. Длинна кадра данных может быть установлена от 4 до 32 бит.

Возможна передача как одиночного, так и множественных кадров.

Для всех случаев необходимо установить напряжение ножки (set pull up) для сигнала SSIx_SS_IN в значение 0. Не рекомендуется смешивать ведомые разных протоколов на одном контроллере.

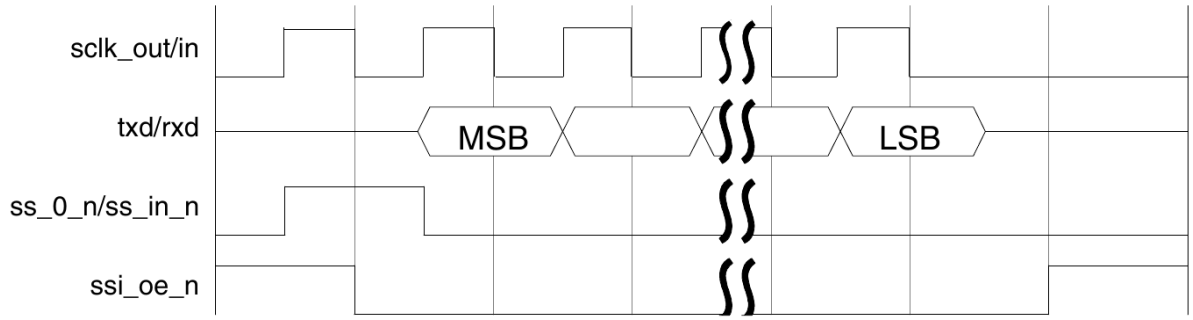


Рисунок 33.4 Пример передачи одного кадра по протоколу SSP

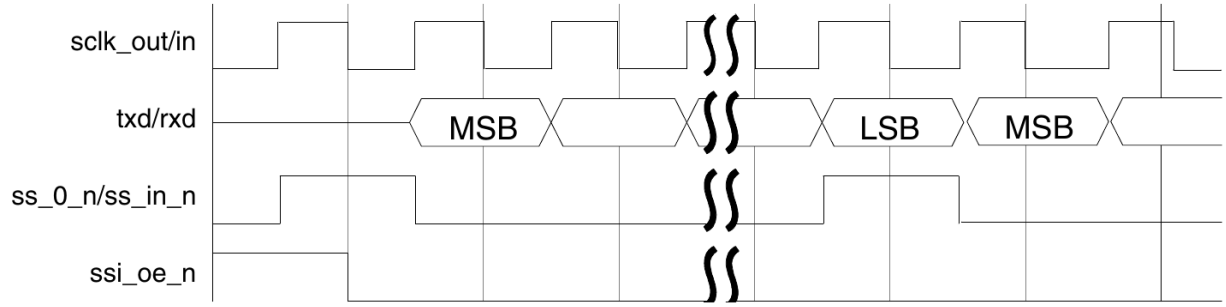


Рисунок 33.5 Пример передачи множественных кадров по протоколу SSP

33.3.6 National Semiconductor Microwire

Обмен данными по протоколу Microwire начинается по заднему фронту сигнала SSIx_SS_IN. Спустя половину периода отправляется первый бит контрольного слова в линию TXD. Длина контрольного слова может изменяться в диапазоне от 1 до 16 бит. Длина контрольного слова настраивается через биты CFS (CTRLR0[15:12]). Во время передачи контрольного слова линия RXD находится в высокоимпедансном состоянии. Направление передачи слова данных контролируется битом MDD (MWCR[1]). Значение MDD = 0 указывает на то, что порт SSI принимает данные от slave-устройства; Значение MDD = 1, указывает на передачу данных slave-устройству. В случае приема данных, через такт после передачи LSB контрольного слова, slave-устройство выдает бит 0 в линию RXD, и после этого следует 4-16 бит слова данных. Данные устанавливаются по заднему фронту линии синхронизации, а выборка по переднему. В случае передачи данных, данные идут сразу за контрольным словом. Сигнал SSIx_SS_IN остается активно-низким на протяжении всей передачи и переходит через полтакта после окончании передачи данных.

Для всех случаев необходимо установить напряжение ножки (set pull up) для сигнала SSIx_SS_IN в значение 1. Не рекомендуется смешивать ведомые разных протоколов на одном контроллере.

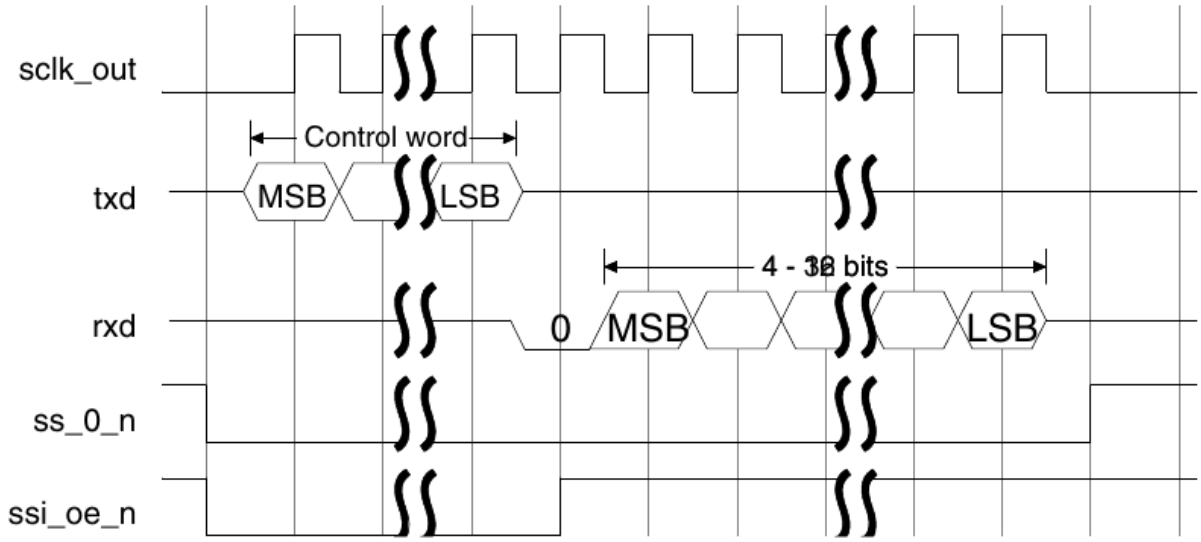


Рисунок 33.6 Пример передачи по протоколу Microwire (MDD = 0)

34. КОНТРОЛЛЕР UART (UART)

34.1 Общая информация

Универсальный асинхронный порт (далее UART) имеет следующие характеристики:

1. FIFO для приема и передачи данных имеют объем по 128 слов.
2. Полностью программируемые параметры последовательного интерфейса: длина символа от 5 до 8 бит, генерация и обнаружение бита четности, генерация стопового сигнала длиной 1, 1.5 или 2 бита.
3. Диагностический режим.
4. Эмуляция символьных ошибок.
5. Аппаратный режим управления потоком данных.
6. Аппаратный интерфейс запросов PDMA.
7. Инфракрасный режим IrDA 1.0 SIR со скоростью до 115.2 КБод (только uart2)
8. Режим RS485(только uart3)

В микросхеме присутствует 4 UART-порта.

34.2 Регистры UART

Перечень регистров uart0_DW_apb_uart приведен в следующей таблице:

Таблица 34.1. Перечень программно-доступных регистров uart0_DW_apb_uart

Смещение	Условное обозначение	Название регистра	Исходное состояние	Тип доступа
0x0	RBR	Регистр буфера приема	0x0	RO
	THR	Регистр передачи данных	0x0	WO
	DLL	Младший регистр делителя частоты	0x0	RW
0x4	DLH	Старший регистр делителя частоты	0x0	RW
	IER	Регистр разрешения прерывания	0x0	RW
0x8	FCR	Регистр управления FIFO	0x0	RW
	IIR	Регистр идентификации прерывания	0x0	RO
0xc	LCR	Регистр управления линией	0x0	RW
0x10	MCR	Регистр управления модемом	0x0	RW
0x14	LSR	Регистр состояния линии	0x0	RO
0x18	MSR	Регистр состояния модема	0x0	RO
0x1c	SCR	Регистр общего назначения	0x0	RW
0x30	SRBR0	Дублирующий регистр буфера приема	0x0	RO
0x34	SRBR1	Дублирующий регистр буфера приема 1	0x0	RO
0x38	SRBR2	Дублирующий регистр буфера приема 2	0x0	RO
0x3c	SRBR3	Дублирующий регистр буфера приема 3	0x0	RO
0x40	SRBR4	Дублирующий регистр буфера приема 4	0x0	RO
0x44	SRBR5	Дублирующий регистр буфера приема 5	0x0	RO
0x48	SRBR6	Дублирующий регистр буфера приема 6	0x0	RO
0x4c	SRBR7	Дублирующий регистр буфера приема 7	0x0	RO
0x50	SRBR8	Дублирующий регистр буфера приема 8	0x0	RO
0x54	SRBR9	Дублирующий регистр буфера приема 9	0x0	RO
0x58	SRBR10	Дублирующий регистр буфера приема 10	0x0	RO
0x5c	SRBR11	Дублирующий регистр буфера приема 11	0x0	RO

Смещение	Условное обозначение	Название регистра	Исходное состояние	Тип доступа
0x60	SRBR12	Дублирующий регистр буфера приема 12	0x0	RO
0x64	SRBR13	Дублирующий регистр буфера приема 13	0x0	RO
0x68	SRBR14	Дублирующий регистр буфера приема 14	0x0	RO
0x6c	SRBR15	Дублирующий регистр буфера приема 15	0x0	RO
0x70	FAR	Регистр доступа FIFO	0x0	RW
0x74	TFR	Регистр чтения FIFO передатчика	0x0	RO
0x78	RFW	Регистр записи FIFO приемника	0x0	RW
0x7c	USR	Регистр статуса UART	0x0	RO
0x80	TFL	Регистр уровня FIFO передатчика	0x0	RO
0x84	RFL	Регистр уровня FIFO приемника	0x0	RO
0x88	SRR	Регистр программного сброса	0x0	RW
0x8c	SRTS	Дублирующий регистр запроса на передачу	0x0	RW
0x90	SBCR	Дублирующий регистр остановки управления	0x0	RW
0x94	SDMAM	Дублирующий регистр режима DMA	0x0	RW
0x98	SFE	Дублирующий регистр активации FIFO	0x0	RW
0x9c	SRT	Дублирующий регистр триггера прерывания приемника	0x0	RW
0xa0	STET	Дублирующий регистр триггера прерывания передатчика	0x0	RW
0xa4	HTX	Регистр остановки передачи	0x0	RW
0xa8	DMASA	Регистр программного оповещения DMA	0x0	RW
0xac	TCR	Регистр управления приемопередатчиком	0x0	RW
0xb0	DE_EN	Регистр управления сигналом DE	0x0	RW
0xb4	RE_EN	Регистр управления сигналом RE	0x0	RW
0xb8	DET	Регистр управления таймером DE	0x0	RW
0xbc	TAT	Регистр управления таймером переключения DE и RE	0x0	RW
0xc0	DLF	Регистр дробной части делителя частоты	0x0	RW
0xf4	CPR	Регистр параметров конфигурации UART	0x0	RO
0xf8	UCV	Регистр версии UART	0x0	RO
0xfc	CTR	Регистр UART ID	0x0	RO

34.2.1 Описание регистра RBR

Регистр буфера приема.

Значение по сбросу: 0x0

Это регистр доступен, только если DLAB бит (LCR[7]) равен 0x0.

Формат регистра RBR приведен в следующей таблице:

Таблица 34.2. Поля регистра RBR

Диапазон	Название	Описание	Сброс
31:8	RSVD_RBR	Резерв	0x0
7:0	RBR	Регистр содержит данные полученные по входному последовательному порту (sin в режиме UART или sir_in в ИК режиме). Данные действительны, если DR бит (LSR[0]) равен 0x1. При выключенном FIFO буфере (FCR[0] равен 0x0), чтение данных из регистра RBR должно	0x0

Диапазон	Название	Описание	Сброс
		происходить до получения следующих данных, иначе данные будут перезаписаны, что приведет к ошибке переполнения. При включенном FIFO буфере (FCR[0] равен 0x1), этот регистр предоставляет доступ к последовательному чтению данных из FIFO буфера приемника. Если FIFO буфер приемника заполнен и чтения данных не произойдет, до получения следующих данных, тогда данные в буфере FIFO не будут перезаписаны, а входящие данные будут потеряны, что, также, вызовет ошибку переполнения.	

34.2.2 Описание регистра THR

Регистр передачи данных.

Значение по сбросу: 0x0

Это регистр доступен, только если DLAB бит (LCR[7]) равен 0x0.

Формат регистра THR приведен в следующей таблице:

Таблица 34.3. Поля регистра THR

Диапазон	Название	Описание	Сброс
31:8	RSVD_RBR	Резерв	0x0
7:0	THR	Регистр содержит данные для отправления по выходному последовательному порту (sout в режиме UART или sir_out_n в ИК режиме). Данные могут быть записаны только, если THRE бит (LSR[5]) равен 0x1. При выключенном FIFO буфере (FCR[0] равен 0x0) и THRE установлен в 0x1, запись символа в регистр THR приводит к установке THRE в 0x0. Запись до установки THRE в 0x1 приведет к перезаписи данных. При включенном FIFO буфере (FCR[0] равен 0x1) и THRE установлен в 0x1, символы могут быть записаны в этот регистр до заполнения FIFO передатчика, данные записанные в случае заполнения FIFO будут утеряны.	0x0

34.2.3 Описание регистра DLL

Младший регистр делителя частоты.

Значение по сбросу: 0x0

Это регистр доступен, только если DLAB бит (LCR[7]) равен 0x1.

Формат регистра DLL приведен в следующей таблице:

Таблица 34.4. Поля регистра DLL

Диапазон	Название	Описание	Сброс
31:8	RSVD_RBR	Резерв	0x0
7:0	DLL	Младшие 8 бит регистра делителя частоты. Скороть передачи данных определяется формулой БОД=Частота UART / (16 * DL), где DL = {DH, DL} В случае если DLL и DLH установлены в 0 передача данных по последовательному порту отключена, после установки значения регистра DLL, должно пройти не меньше 8 периодов тактовой частоты UART	0x0

34.2.4 Описание регистра DLH

Старший регистр делителя частоты.

Значение по сбросу: 0x0

Это регистр доступен, только если DLAB бит (LCR[7]) равен 0x1.

Формат регистра DLH приведен в следующей таблице:

Таблица 34.5. Поля регистра DLH

Диапазон	Название	Описание	Сброс
31:8	RSVD_RBR	Резерв	0x0
7:0	DLL	Старшие 8 бит регистра делителя частоты. Скороть передачи данных определяется формулой БОД=Частота UART / (16 * DL), где DL = {DH, DL} В случае если DLL и DLH установлены в 0 передача данных по последовательному порту отключена, после установки значения регистра DLH, должно пройти не меньше 8 периодов тактовой частоты UART	0x0

34.2.5 Описание регистра IER

Регистр разрешения прерывания

Значение по сбросу: 0x0.

Это регистр доступен, только если DLAB бит (LCR[7]) равен 0x0.

Формат регистра IER приведен в следующей таблице:

Таблица 34.6. Поля регистра IER

Диапазон	Название	Описание	Сброс
31:8	RSVD_IER_31to8	Резерв	0x0

Диапазон	Название	Описание	Сброс
7	PTIME	Режим программируемого порога для прерывания. Используется для включения/отключения прерываний по заданному порогу. 0x0 – отключен 0x1 – включен	0x0
6:5	RSVD_IER_6to5	Резерв	0x0
4	ELCOLR	Управление методом сброса статуса в регистре LSR. Доступен только в случае ошибок: переполнения, четности, кадрирования также в состоянии остановки прерывания. 0x0 – Поле статуса регистра LSR are cleared или по чтению Rx FIFO или по чтению регистра LSR. 0x1 – Поле статуса сбрасываются только по чтению регистра LSR.	0x0
3	EDSSI	Разрешение прерывания по статусу модема. Четвертое по приоритету прерываний. 0x1 – разрешено 0x0 – запрещено	0x0
2	ELSI	Разрешение прерывания по состоянию линии приема. Первое по приоритету прерываний. 0x1 – разрешено 0x0 – запрещено	0x0
1	ETBEI	Разрешение прерывания по регистру THRE. Третье по приоритету прерываний 0x1 – разрешено 0x0 – запрещено	0x0
0	ERBFI	Разрешение прерывания по доступности полученных данных (Enable Received Data Available Interrupt) или, при в включенном FIFO, прерывания по тайм-ауту входных данных (Character Timeout Interrupt). Второе по приоритету прерываний. 0x1 – разрешено 0x0 – запрещено	0x0

34.2.6 Описание регистра FCR

Регистр управления FIFO

Значение по сбросу: 0x0.

Этот регистр доступен только в том случае, если выбрана настройка с FIFO (FIFO_MODE != NONE). Если FIFO не используется, этот регистр не существует, и запись на этот адрес регистра не будет иметь никакого эффекта.

Формат регистра FCR приведен в следующей таблице:

Таблица 34.7. Поля регистра FCR

Диапазон	Название	Описание	Сброс
31:8	RSVD_IIR_31to8	Резерв	0x0
7:6	RT	Порог прерывания FIFO приемника. В режиме auto flow control, определяет, когда будет сброшен сигнал rts_n, только если RTC_FCT выключен. Также определяет, когда сигнал dma_rx_req_n будет установлен, в некоторых режимах работы. 0x0 – 1 запись в FIFO 0x1 – FIFO заполнен на ¼ 0x2 – FIFO заполнен на ½ 0x3 – FIFO заполнен без двух записей	0x0
5:4	TET	Управление уровнем прерывания THRE Запись не имеет эффекта если THRE_MODE_USER == Disabled Определяет уровень срабатывания THRE прерывания. Также, определяет установку сигнала dma_tx_req_n в некоторых режимах работы. 0x0 – FIFO пуст 0x1 – 2 записи в FIFO 0x2 – FIFO заполнен на ¼ 0x3 – FIFO заполнен на ½	0x0
3	DMAM	Резерв	0x0
2	XFIFOR	Сброс FIFO передатчика. Очищает FIFO передатчика. Также сбрасывает DMA TX запрос. Бит сбрасывается после записи.	0x0
1	RFIFOR	Сброс FIFO приемника. Очищает FIFO приемника. Также сбрасывает DMA RX запрос.	0x0
0	FIFOE	Управление FIFO При любом изменении этого боля, FIFO приемника и передатчика очищаются. 0x0 – FIFO приемника и передатчика (XMIT, RCVR) отключены 0x1 – FIFO приемника и передатчика (XMIT, RCVR) включены	0x0

34.2.7 Описание регистра IIR

Регистр идентификации прерывания

Этот регистр доступен всегда.

Формат регистра IIR приведен в следующей таблице:

Таблица 34.8. Поля регистра IIR

Диапазон	Название	Описание	Сброс
31:8	RSVD_IIR_31to8	Резерв	0x0
7:6	FIFOSE	Индикация рабочего состояния FIFO. 0x0 – отключен 0x1 – включен	0x0
5:4	RSVD_IIR_5to4	Резерв	0x0
3:0	IID	ID прерывания (IID). Отображает прерывание с наивысшим приоритетом 0x0110 – статус линии приемника 0x1000 – доступны данные на принимающей линии 0x1100 – таймаут ожидания приемника 0x0010 – прерывание THRE 0x0000 – статус модема 0x0111 – определение попытки записи	0x1

34.2.8 Описание регистра LCR

Регистр управления линией

Формат регистра LCR приведен в следующей таблице:

Таблица 34.9. Поля регистра LCR

Диапазон	Название	Описание	Сброс
31:8	RSVD_LCR_31to8	Резерв	0x0
7	DLAB	Бит доступа к делителю частоты, используется для разрешения чтения и записи регистров DLL/LPDLL и LPDLH. Данный бит должен быть установлен в 0x0 после настройки частоты интерфейса UART, для доступа к другим регистрам. Доступен только для записи в случае если USR[0] == 0x0 0x0 – отключен 0x1 – включен	0x0
6	BC	Бит обрыва линии. 0x0 – нормальная работа 1x1 – на выходе sout устанавливается низкий уровень. В ИК режиме линия sir_out_n продолжительно пульсирует. В режиме кольцевой связи линия sir_out_n в состоянии низкого уровня.	0x0
5	SP	Бит установки паритета. Доступен для записи, когда USR[0] не 0, используется для установки паритета.	0x0

Диапазон	Название	Описание	Сброс
		0x0 – SP отключен 0x1 - Когда PEN, EPS установлены в 1, бит паритета передается и принимается как логический 0. Если PEN установлен в 1, а EPS в 0, то бит паритета передается и принимается как логическая 1.	
4	EPS	Бит четности паритета Доступен для записи, когда USR[0] не 0, используется для установки четности паритета. 0x0 – нечетность 0x1 – четность	0x0
3	PEN	Разрешение паритета Доступен для записи, когда USR[0] не 0, используется для разрешения приема, проверки и передачи контрольного бита (бита паритета).	0x0
2	STOP	Количество стоп-битов Доступен для записи, когда USR[0] не 0. 0x0 – 1 стоп-бит, 0x1 – 1.5 стоп-бита, если LCR[0:1]=0, иначе 2 стоп-бита. Приемник анализирует только первый стоп бит.	0x0
1:0	DLS	Количество бит данных в передаваемом символе. Доступен для записи, когда USR[0] не 0. 0x00 – 5 бит 0x01 – 6 бит 0x10 – 7 бит 0x11 – 8 бит	0x0

34.2.9 Описание регистра MCR

Регистр управления модемом.

Формат регистра MCR приведен в следующей таблице:

Таблица 34.10. Поля регистра MCR

Диапазон	Название	Описание	Сброс
31:7	RSVD_MCR_31to7	Резерв	0x0
6	SIRE	Активация ИК режима	0x0
5	AFCE	Включение автоматического контроля управления потоком возможно только при включенных FIFO (FCR[0]=1). 0x0 – Автоматическое управление потоком выключено.	0x0

Диапазон	Название	Описание	Сброс
		0x1 – Автоматическое управление потоком включено.	
4	LoopBack	Активация петлевого режима Используется для активации диагностического режима с целью тестирования. При отключенном ИК режиме, линия sout, будут находиться в высоком состоянии, и подключена на линию sin, до отключения петлевого режима. В этом режиме функционируют все прерывания. Также, все модемные и программируемые выходы будут отключены и закольцованы на модемные входы. В ИК режиме линия sir_out_n будет находиться в низком уровне и закольцована на линию sir_in.	0x0
3	OUT2	Программируемый выход 2 Значение, записанное в поле, инвертируется и передается на выход	0x0
2	OUT1	Программируемый выход 1 Значение, записанное в поле, инвертируется и передается на выход	0x0
1	RTS	Запрос на отправление данных. Этот бит используется для управления выводом RTS_N. Вывод RTS_N используется для информирования модема оконечного устройства о том, что UART готов к обмену данными. При выключенном режиме автоматического управления потоком (MCR[5] = 0), сигнал RTS_N переводится в низкий уровень записью 1 в бит RTS, и переводится в высокий уровень записью 0 в бит RTS. При включенном режиме автоматического управления потоком (MCR[5]=1), и включенных FIFO (FCR[0]=1), выходной сигнал RTS_N может управляться таким же способом, но он также меняет свое значение в зависимости от уровня заполнения FIFO приемника (становится равным 1, если кол-во записей в FIFO приемника превышает пороговое значение заполнения FIFO приемника, и переходит обратно в 0, когда кол-во записей в FIFO становится меньше этого значения).	0x0
0	DTR	Управление линией dtr_n Используется для прямого управления сигналом dtr_n. Сигнал DTR используется для оповещения модема, о готовности абонента установить соединение.	0x0

34.2.10 Описание регистра LSR

Регистр состояния линии

Формат регистра LSR приведен в следующей таблице:

Таблица 34.11. Поля регистра LSR

Диапазон	Название	Описание	Сброс
31:9	RSVD_LSR_31to9	Резерв	0x0
8	RSVD_ADDR_RCVD	Резерв	0x0
7	RFE	Бит ошибки приемника Определяет наличие хотя бы одной ошибки четности, кадрирования или состояние обрыва. Сбрасывается при чтении LSR.	0x0
6	TEMT	Бит отсутствия передаваемых данных Устанавливается в 1 в случае отсутствия данных в FIFO буфере и сдвиговом регистре передатчика.	0x1
5	THRE	Бит отсутствия данных в буфере передатчика Показывает, что UART готов принять следующий символ для передачи. Устанавливается, когда содержимое регистра THR передается в передающий сдвигающий регистр. Одновременно с этим генерируется прерывание THREI, если оно разрешено. Бит сбрасывается при записи символа в регистр THR.	0x1
4	BI	Обрыв линии Устанавливается, если вход приема данных находится в состоянии 0 не менее чем время передачи всего символа. В режиме FIFO этот бит показывает на ошибку в символе, находящемся наверху FIFO. При возникновении этой ситуации, в FIFO загружается только один нулевой символ. Прием следующих символов разрешается после того, как вход приема данных перейдет в единицу и будет принят действительный стартовый бит. Бит сбрасывается при чтении содержимого регистра LSR.	0x0
3	FE	Ошибка кадрирования. Устанавливается, если стоп-бит равен нулю. В режиме FIFO этот бит показывает на ошибку в символе, находящемся наверху FIFO. После этой ошибки UART пере синхронизируется. Бит сбрасывается при чтении содержимого регистра LSR.	0x0

Диапазон	Название	Описание	Сброс
2	PE	Ошибка паритета (контрольного бита) В режиме FIFO этот бит показывает на ошибку в символе, находящемся наверху FIFO. Бит сбрасывается при чтении содержимого регистра LSR.	0x0
1	OE	Ошибка переполнения Устанавливается, если содержимое регистра RBR не было прочитано, в сдвигающий регистр принят следующий символ и начат прием очередного символа. При этом новый символ записывается в сдвигающий регистр вместо старого. В режиме FIFO устанавливается, если после перехода порогового уровня FIFO заполнено до конца, во входной сдвигающий регистр полностью принят следующий символ и начат прием очередного символа. При этом в FIFO ничего не передается. Бит сбрасывается при чтении содержимого регистра LSR.	0x0
0	DR	Бит готовности данных Показывает наличие данных в приемнике, которые еще не были прочитаны.	0x0

34.2.11 Описание регистра MSR

Регистр состояния модема.

При записи в биты 0, 1, 2 или 3 логической 1, для отображения изменение настроек модема, будет сгенерирован прерывания по статусу модема (если он разрешен), независимо от того когда произошла запись. Биты (бит 0, 1, 3) могут быть установлены после сброса, даже если их соответствующие модемные сигналы неактивны, поскольку синхронизированная версия модемных сигналов имеет значение сброса 0 и изменяется на значение 1 после сброса. Для предотвращения нежелательных прерываний, чтение MSR можно производить после сброса.

Формат регистра MSR приведен в следующей таблице:

Таблица 34.12. Поля регистра MSR

Диапазон	Название	Описание	Сброс
31:8	RSVD_MSR_31to8	Резерв	0x0
7	DCD	Обнаружение носителя данных Используется для чтения состояния линии dcd_n.	0x0
6	RI	Индикатор доступа Используется для чтения состояния линии ri_n.	0x0

Диапазон	Название	Описание	Сброс
5	DSR	Готовность данных Используется для чтения состояния линии dsr_n.	0x0
4	CTS	Готовность отправки Используется для чтения состояния линии cts_n.	0x0
3	DDCD	Обнаружение изменения линии dcd_n Определяет изменялась ли линия dcd_n после последнего чтения регистра MSR.	0x0
2	TERI	Обнаружение изменения линии ri_n Определяет изменялась ли линия ri_n после последнего чтения регистра MSR.	0x0
1	DDSR	Обнаружение изменения линии dsr_n Определяет изменялась ли линия dsr_n после последнего чтения регистра MSR.	0x0
0	DCTS	Обнаружение изменения линии cts_n Определяет изменялась ли линия cts_n после последнего чтения регистра MSR.	0x0

34.2.12 Описание регистра SCR

Регистр общего назначения

Формат регистра SCR приведен в следующей таблице:

Таблица 34.13. Поля регистра SCR

Диапазон	Название	Описание	Сброс
31:8	RSVD_SCR_31to8	Резерв	0x0
7:0	SCR	Этот регистр предназначен для временного хранения данных. Не имеет определенной цели в управлении контроллером.	0x0

34.2.13 Описание регистра SRBR0 - 15

Дублирующий регистр буфера приема

Формат регистров SRBR0 - 15 приведен в следующей таблице:

Таблица 34.14. Поля регистров SRBR0 - 15

Диапазон	Название	Описание	Сброс
31:8	RSVD_SRBRn	SRBR0 31 to SRBRN_REG_SIZE Reserved bits read read as 0.	0x0
7:0	SRBRn	Этот регистр дублирует регистр RBR. К нему можно обратиться по одному из 32-х адресов.	0x0

34.2.14 Описание регистра FAR

Регистр доступа FIFO

Формат регистра FAR приведен в следующей таблице:

Таблица 34.15. Поля регистра FAR

Диапазон	Название	Описание	Сброс
31:1	RSVD_FAR_31to1	Резерв	0x0
0	FAR	Регистр доступа FIFO Позволяет разрешить прямой доступ к FIFO буферам.	0x0

34.2.15 Описание регистра TFR

Регистр чтения FIFO передатчика

Формат регистра TFR приведен в следующей таблице:

Таблица 34.16. Поля регистра TFR

Диапазон	Название	Описание	Сброс
31:8	RSVD_TFR_31to8	TFR 31to8 Reserved bits read as 0.	0x0
7:0	TFR	Чтение данных FIFO передатчика Доступен в режиме доступа к FIFO (FAR[0] = 1).	0x0

34.2.16 Описание регистра RFW

Регистр записи FIFO приемника

Формат регистра RFW приведен в следующей таблице:

Таблица 34.17. Поля регистра RFW

Диапазон	Название	Описание	Сброс
31:10	RSVD_RFW_31to10	Резерв	0x0

Диапазон	Название	Описание	Сброс
9	RFFE	Запись ошибки кадрирования При отключённом режиме FIFO, записывает ошибку в регистр RBR.	0x0
8	RFPE	Запись ошибки четности При отключённом режиме FIFO, записывает ошибку в регистр RBR.	0x0
7:0	RFWD	Запись в FIFO приемника Доступен в режиме доступа к FIFO (FAR[0] = 1).	0x0

34.2.17 Описание регистра USR

Регистр статуса UART

Формат регистра USR приведен в следующей таблице:

Таблица 34.18. Поля регистра USR

Диапазон	Название	Описание	Сброс
31:5	RSVD_USR_31to5	Резерв	0x0
4	RFF	FIFO приемника полон 0x0 – FIFO приемника не полон 1x1 – FIFO приемника полон	0x0
3	RFNE	FIFO приемника не пуст 0x0 – FIFO приемника не пуст 0x1 – FIFO приемника пуст	0x0
2	TFE	FIFO передатчика пуст 0x0 – FIFO передатчика не пуст 0x1 – FIFO передатчика пуст	0x1
1	TFNF	FIFO передатчика не полон 0x0 – FIFO передатчика полон 0x1 – FIFO передатчика не полон	0x1
0	BUSY	UART занят. Этот бит установлен в 1, в следующих случаях: 1. В настоящее время происходит передача данных по последовательному интерфейсу. 2. Регистр THR содержит данные на передачу, при не нулевом значении делителя частоты ({DLH,DLL} ~= 0) и LCR.DLAB = 0. 3. В настоящее время происходит прием данных по последовательному интерфейсу. 4. Регистр RBR содержит данные на прием.	0x0

34.2.18 Описание регистра TFL

Регистр уровня FIFO передатчика

Формат регистра TFL приведен в следующей таблице:

Таблица 34.19. Поля регистра TFL

Диапазон	Название	Описание	Сброс
31:6	RSVD_TFL_31toADDR_WIDTH	Резерв	0x0
5:0	tfl	Уровень FIFO передатчика Отображает количество слов данных в FIFO передатчика.	0x0

34.2.19 Описание регистра RFL

Регистр уровня FIFO приемника

Таблица 34.20. Поля регистра RFL

Диапазон	Название	Описание	Сброс
31:6	RSVD_RFL_31toADDR_WIDTH	Резерв	0x0
5:0	rfl	Уровень FIFO приемника Отображает количество слов данных в FIFO приемника.	0x0

34.2.20 Описание регистра SRR

Регистр программного сброса

Формат регистра SRR приведен в следующей таблице:

Таблица 34.21. Поля регистра SRR

Диапазон	Название	Описание	Сброс
31:3	RSVD_SRR_31to3	Резерв	0x0
2	XFR	Дублирует FCR[2]	0x0
1	RFR	Дублирует FCR[1]	0x0
0	UR	Программный сброс UART	0x0

34.2.21 Описание регистра SRTS

Дублирующий регистр запроса на передачу

Формат регистра SRTS приведен в следующей таблице:

Таблица 34.22. Поля регистра SRTS

Диапазон	Название	Описание	Сброс
31:1	RSVD_SRTS_31to1	Резерв	0x0
0	SRTS	Дублирует MCR[1]	0x0

34.2.22 Описание регистра SBCR

Дублирующий регистр остановки управления

Формат регистра SBCR приведен в следующей таблице:

Таблица 34.23. Поля регистра SBCR

Диапазон	Название	Описание	Сброс
31:1	RSVD_SBCR_31to1	Резерв	0x0
0	SBCB	Дублирует LCR[6]	0x0

34.2.23 Описание регистра SDMAM

Дублирующий регистр режима DMA

Формат регистра SDMAM приведен в следующей таблице:

Таблица 34.24. Поля регистра SDMAM

Диапазон	Название	Описание	Сброс
31:1	RSVD_SDMAM_31to1	Резерв	0x0
0	SDMAM	Дублирует FCR[3]	0x0

34.2.24 Описание регистра SFE

Дублирующий регистр активации FIFO

Формат регистра SFE приведен в следующей таблице:

Таблица 34.25. Поля регистра SFE

Диапазон	Название	Описание	Сброс
31:1	RSVD_SFE_31to1	Резерв	0x0
0	SFE	Дублирует FCR[0]	0x0

34.2.25 Описание регистра SRT

Дублирующий регистр триггера прерывания приемника

Формат регистра SRT приведен в следующей таблице:

Таблица 34.26. Поля регистра SRT

Диапазон	Название	Описание	Сброс
31:2	RSVD_SRT_31to2	Резерв	0x0
1:0	SRT	Дублирует FCR[7:6]	0x0

34.2.26 Описание регистра STET

Дублирующий регистр триггера прерывания передатчика

Формат регистра STET приведен в следующей таблице:

Таблица 34.27. Поля регистра STET

Диапазон	Название	Описание	Сброс
31:2	RSVD_STET_31to2	Резерв	0x0
1:0	STET	Дублирует FCR[5:4]	0x0

34.2.27 Описание регистра HTX

Регистр остановки передачи

Формат регистра HTX приведен в следующей таблице:

Таблица 34.28. Поля регистра HTX

Диапазон	Название	Описание	Сброс
31:1	RSVD_HTX_31to1	Резерв	0x0
0	HTX	Остановка передачи Используется для остановки передачи в целях тестирования, таким образом, что буфер FIFO может быть заполнен. В случае, если FIFO режим отключен запись не в этот регистр не повлияет на работу UART.	0x0

34.2.28 Описание регистра DMASA

Регистр программного оповещения DMA

Формат регистра DMASA приведен в следующей таблице:

Таблица 34.29. Поля регистра DMASA

Диапазон	Название	Описание	Сброс
31:1	RSVD_DMASA_31to1	Резерв	0x0
0	DMASA	Программное оповещения DMA Бит самосбрасывается и не требует ручного сброса.	0x0

Диапазон	Название	Описание	Сброс

34.2.29 Описание регистра TCR

Регистр управления приемопередатчиком

Этот регистр используется, для включения/отключения режима RS485, также для управления полярностью сигналов de и re.

Формат регистра TCR приведен в следующей таблице:

Таблица 34.30. Поля регистра TCR

Диапазон	Название	Описание	Сброс
31:5	RSVD_TCR_31to5	Резерв	0x0
4:3	XFER_MODE	Режим передачи 0x0 – одновременная передача и прием 0x1 – DE и RE взаимно исключающие сигналы, ожидается, что при программировании, только один из этих сигналов будет в активном состоянии. Во время переключения, будет задействован регистр TAT для определения задержек. 0x2 – DE и RE взаимно исключающие сигналы, изначально находится в состоянии приема (зависит от состояний DE_EN/RE_EN) при появлении данных в передающем буфере, UART ожидает окончания приема, после чего переключается на передачу данных, до тех пока буфер передачи не будет пуст.	0x0
2	DE_POL	Полярность сигнала DE 0x0 – активное высокое состояние 0x1 – активное низкое состояние	0x0
1	RE_POL	Полярность сигнала RE 0x0 – активное высокое состояние 0x1 – активное низкое состояние	0x0
0	RS485_EN	Режим RS485 0x0 – отключен, все поля этого регистра, также регистры DE_EN, RE_EN, TAT являются зарезервированными. 0x1 – включен все регистры для настройки режима RS485 находятся в рабочем режиме.	0x0

34.2.30 Описание регистра DE_EN

Регистр управления сигналом DE

Формат регистра DE_EN приведен в следующей таблице:

Таблица 34.31. Поля регистра DE_EN

Диапазон	Название	Описание	Сброс
31:1	RSVD_DE_EN_31to1	Резерв	0x0
0	DE_Enable	Регистр управления сигналом DE 0x0 – сбросить 0x1 – установить	0x0

34.2.31 Описание регистра RE_EN

Регистр управления сигналом RE

Формат регистра RE_EN приведен в следующей таблице:

Таблица 34.32. Поля регистра RE_EN

Диапазон	Название	Описание	Сброс
31:1	RSVD_RE_EN_31to1	Резерв	0x0
0	RE_Enable	Регистр управления сигналом RE 0x0 – сбросить 0x1 – установить	0x0

34.2.32 Описание регистра DET

Регистр управления таймером DE

Формат регистра DET приведен в следующей таблице:

Таблица 34.33. Поля регистра DET

Диапазон	Название	Описание	Сброс
31:24	RSVD_DE_DEAT_31to24	Резерв	0x0
23:16	DE_De-assertion_Time	Управления таймером сброса DE Это поле определяет количество периодов, тактового сигнала между окончанием приема данных и сброса сигнала DE.	0x0
15:8	RSVD_DE_AT_15to8	Резерв	0x0
7:0	DE_Assertion_Time	Управления таймером установки DE Это поле определяет количество периодов, тактового сигнала между установкой сигнала DE и началом передачи данных.	0x0

34.2.33 Описание регистра TAT

Регистр управления таймером переключения DE и RE

Формат регистра TAT приведен в следующей таблице:

Таблица 34.34. Поля регистра TAT

Диапазон	Название	Описание	Сброс
31:16	RE_to_DE	Задержка переключения с RE на DE Если DET=0, тогда задержка устанавливается в виде запрограммированного значения + 3. Если DET=1, тогда задержка устанавливается в виде запрограммированного значения + 2. Если DET>1, тогда задержка устанавливается в виде запрограммированного значения + 1.	0x0
15:0	DE_to_RE	Задержка переключения с RE на DE Задержка устанавливается в виде запрограммированного значения + 1.	0x0

34.2.34 Описание регистра DLF

Регистр дробной части делителя частоты

Дробное значение вычисляется с помощью формулы $\frac{DLF}{16}$

Значение DLF задает числитель дроби

Пример.

DLF = 0010_2 => 2_{10} => числитель дроби = 2 => $\frac{2}{16}$ => 0,125

Пример для других значений DLF приведен в следующей таблице:

Таблица 35. Пример расчета дробного значения

Значение DLF	Дробь	Дробное значение
0000	$\frac{0}{16}$	0,0000
0001	$\frac{1}{16}$	0,0625
0010	$\frac{2}{16}$	0,125
0011	$\frac{3}{16}$	0,1875
0100	$\frac{4}{16}$	0,25
0101	$\frac{5}{16}$	0,3125
0110	$\frac{6}{16}$	0,375
0111	$\frac{7}{16}$	0,4375
1000	$\frac{8}{16}$	0,5
1001	$\frac{9}{16}$	0,5625
1010	$\frac{10}{16}$	0,625

Значение DLF	Дробь	Дробное значение
1011	$\frac{11}{16}$	0,6875
1100	$\frac{12}{16}$	0,75
1101	$\frac{13}{16}$	0,8125
1110	$\frac{14}{16}$	0,875
1111	$\frac{15}{16}$	0,9375

Формат регистра DLF приведен в следующей таблице:

Таблица 34.36. Поля регистра DLF

Диапазон	Название	Описание	Сброс
31:4	RSVD_DLF	Резерв	0x0
3:0	DLF	Дробная часть делителя частоты Определяется как $\frac{DLF}{16}$ Значение по сбросу: 0x0	0x0

34.2.35 Описание регистра CPR

Регистр параметров конфигурации UART

Формат регистра CPR приведен в следующей таблице:

Таблица 34.37. Поля регистра CPR

Диапазон	Название	Описание	Сброс
31:24	RSVD_CPR_31to24	Резерв	0x0
23:16	FIFO_MODE	Encoding of FIFO_MODE configuration parameter value.	0x2
15:14	RSVD_CPR_15to14	Резерв	0x0
13	DMA_EXTRA	Encoding of DMA_EXTRA configuration parameter value.	0x1
12	UART_ADD_ENCODED_PARAMS	Encoding of UART_ADD_ENCODED_PARAMS configuration parameter value.	0x1
11	SHADOW	Encoding of SHADOW configuration parameter value.	0x1
10	FIFO_STAT	Encoding of FIFO_STAT configuration parameter value.	0x1
9	FIFO_ACCESS	Encoding of FIFO_ACCESS configuration parameter value.	0x1

Диапазон	Название	Описание	Сброс
8	ADDITIONAL_FEAT	Encoding of ADDITIONAL_FEATURES configuration parameter value.	0x1
7	SIR_LP_MODE	Encoding of SIR_LP_MODE configuration parameter value.	0x0
6	SIR_MODE	Encoding of SIR_MODE configuration parameter value.	0x0
5	THRE_MODE	Encoding of THRE_MODE configuration parameter value.	0x1
4	AFCE_MODE	Encoding of AFCE_MODE configuration parameter value.	0x1
3:2	RSVD_CPR_3to2	Резерв	0x0
1:0	APB_DATA_WIDTH	Encoding of APB_DATA_WIDTH configuration parameter value.	0x2

34.2.36 Описание регистра UCV

Регистр версии UART

Формат регистра UCV приведен в следующей таблице:

Таблица 34.38. Поля регистра UCV

Диапазон	Название	Описание	Сброс
31:0	UART_Component_Version	ASCII значение Для каждого номера версии следует знак *. Например: 32_30_31_2A представляется в виде 2.01*.	0x3430312a

34.2.37 Описание регистра CTR

Регистр UART ID

Формат регистра CTR приведен в следующей таблице:

Таблица 34.39. Поля регистра CTR

Диапазон	Название	Описание	Сброс
31:0	Peripheral_ID	Периферийное ID	0x44570110

34.3 Функционирование порта

34.3.1 Последовательный протокол (RS232)

Так как последовательный обмен между контроллером UART и оконечным устройством является асинхронным, то к последовательным данным добавляются дополнительные биты Start и Stop, обозначающие начало и конец передачи. Совокупность слова данных и битов Start и Stop называется символом. Структура символа данных отображена на Рисунок 34. 1.

Также к символу может быть добавлен бит контроля четности (Parity). Этот бит появляется

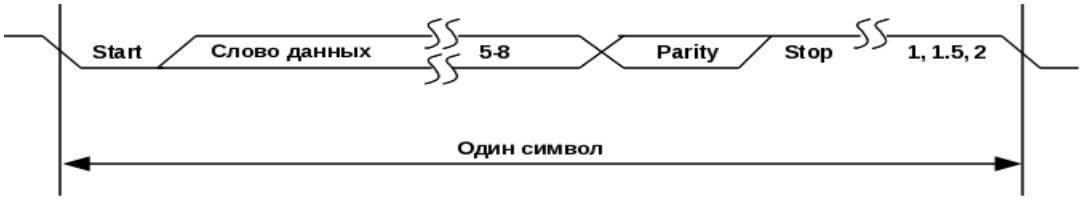


Рисунок 34. 1. RS232. Формат символа данных

после последнего бита данных и перед Stop битом, чтобы предоставить UART возможность выполнять простую проверку ошибок в полученных данных.

Управление параметрами символа последовательного обмена происходит через регистр LCR. Отдельные биты слова данных отправляются после Start бита, начиная с младшего значащего бита (LSB). За ними следует дополнительный бит четности, за которым следуют Stop биты, которые могут быть 1, 1.5, 2.

Все биты передаются в течение одинакового времени (исключение из этого правила является бит половинного Stop, когда используется 1,5 Stop бита). Эта продолжительность называется битовым периодом или битовым временем; один бит времени равняется шестнадцати тактам бод.

Типовой сигнал данных представляет собой просто напряжение, уровень которого изменяется между состояниями логического нуля и единицы. Приемник может правильно преобразовать эти логические состояния в цифровые данные, только если он знает, когда сделать выборку сигнала.

Передатчик формирует битовый поток на основе своего тактового сигнала, а целью приемника является использование своего тактового сигнала для выборки входящих данных в середине каждого битового периода.

Последовательность приемника начинается с заднего фронта стартового бита. Внутренний тактовый генератор приемника полностью независим от внутреннего тактового генератора передатчика – другими словами, эта граница спада сигнала может соответствовать любой точке в тактовом сигнале приемника.

Допустим, что один битовый период соответствует 16 циклам тактового сигнала приемника. В этом случае синхронизация и выборка могут быть выполнены следующим образом:

1. Процесс приема запускается по границе спада стартового бита.
2. Приемник ждет в течение 8 циклов тактового сигнала, чтобы установить момент выборки, который находится близко к середине периода бита.
3. Приемник ждет в течение 16 циклов тактового сигнала, которые приводят его к середине периода первого бита данных.
4. Первый бит данных оцифровывается и сохраняется в регистре приемника, а затем модуль снова ожидает 16 циклов тактового сигнала перед выборкой второго бита данных.
5. Этот процесс повторяется до тех пор, пока все биты данных не будут выбраны и сохранены, а затем нарастающий фронт стопового бита возвращает интерфейс UART в режим ожидания.

На Рисунок 34. 2 показаны точки выборки первых двух битов в последовательном символе.

В рамках стандарта, дополнительный выходной сигнал тактовой частоты в бодах

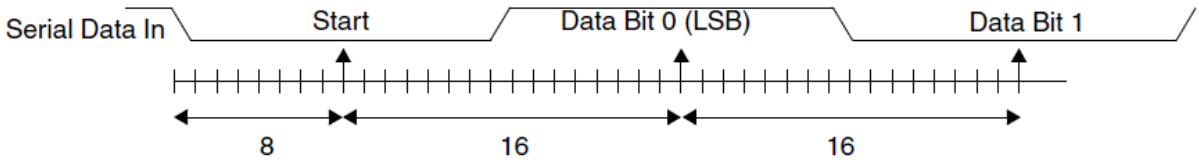


Рисунок 34. 2. Точки выборки данных

(baudout_n) предоставляет информацию о синхронизации принимающим устройствам, которым это требуется. Скорость передачи данных UART управляется sclk или pclk в реализации с одним тактом, а также DLH и DLL.

На Рисунок 34. **3** показана временная диаграмма выхода baudout для различных значений делителя.

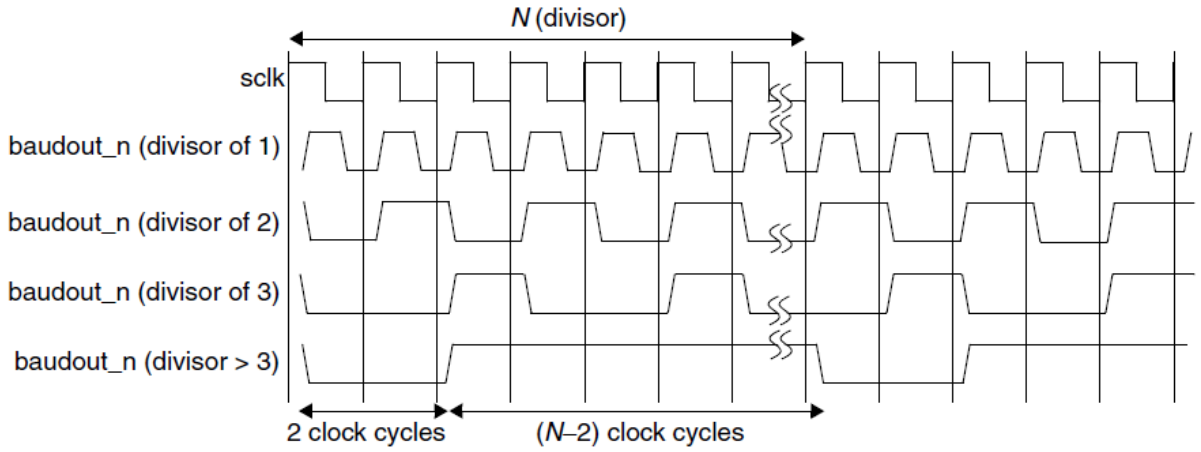


Рисунок 34. 3. Временная диаграмма опорного тактового сигнала в бодах

34.3.2 Последовательный протокол (RS485)

Стандарт RS485 поддерживает последовательную связь по конфигурации с витой парой, такой как RS232. Разница между стандартами RS232 и RS485 заключается в использовании симметричной линии для передачи. Это использование также известно как дифференциальный формат, который отправляет один и тот же сигнал по двум отдельным линиям с фазовой задержкой, а затем сравнивает сигналы в конце, вычитает любой шум и складывает их, чтобы восстановить уровень сигнала. Этот процесс позволяет стандарту RS485 быть жизнеспособным на больших расстояниях.

UART поддерживает последовательный протокол RS485, который обеспечивает передачу последовательных данных с использованием интерфейса RS485. Сигналы включения драйвера (DE) и включения приемника (RE) генерируются для включения поддержки интерфейса RS485. Сигналы DE и RE генерируются аппаратно, и время подтверждения/отключения для этих сигналов программируется. Активный уровень этих сигналов настраивается.

Конфигурация UART для интерфейса RS485 выполняет следующие действия:

1. Бит 0 регистра TCR включает или отключает режим RS485.
2. Бит 1 и бит 2 TCR используются для выбора полярности сигналов RE и DE.
3. Биты TCR [4:3] выбирают тип передачи в режиме RS485.
4. Регистры DE_EN и RE_EN используются для программного управления сигналами DE и RE.

5. Регистр DET используется для программирования времени подтверждения и отмены сигнала DE.

6. Регистр TAT используется для программирования времени выполнения с DE на RE и RE на DE.

34.3.3.1 Время подтверждения и отмены DE

Время подтверждения и отмены сигнала DE контролируется через регистр DET:

- Время утверждения DE (DET[7:0]): время утверждения – это время между активацией сигнала DE и началом Start битом.

- Время отмены подтверждения DE (DET[15:8]): Время отмены подтверждения – это время между концом последнего Stop бита в переданном символе и отключением сигнала DE.

Аппаратное обеспечение гарантирует соблюдение этих значений для подтверждения DE и отмены подтверждения DE до/после активации передачи данных.

На Рисунок 34. 4 t1 – представляет время подтверждения DE, а t2 — время отмены подтверждения DE. DE отменяется только после того, как все символы данных переданы.

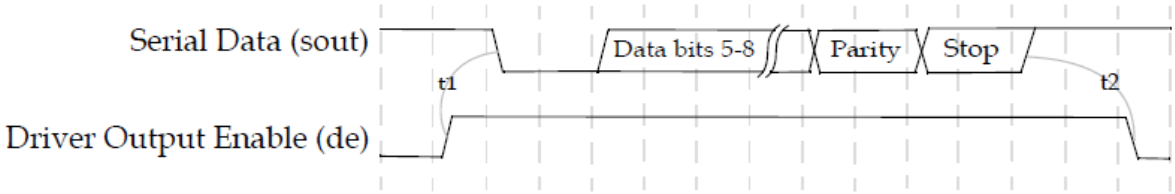


Рисунок 34. 4. DE утверждение и отказ от утверждения

34.3.3.2 Режимы RS485

UART состоит из следующих режимов RS485, основанных на XFER_MODE в регистре TCR:

- Полнодуплексный режим — XFER_MODE установлен в 0.

- Программно-управляемый полудуплексный режим — XFER_MODE установлен в 1.

- Полудуплексный режим с аппаратным управлением — XFER_MODE установлен в 2.

34.3.3.2.1 Полнодуплексный режим

Полнодуплексный режим поддерживает одновременную передачу и прием.

Сигнал DE:

- Становится активным, если выполняются оба эти условия:
    - Если для DE_EN[0] установлено значение 1.
    - Если THR не пуст в режиме без FIFO или FIFO передатчика не пуст в режиме FIFO.
- Становится неактивным, если выполняются оба эти условия:
    - Когда текущая последовательная передача завершена.
    - Либо DE_EN[0] установлен в 0, либо передатчик FIFO пуст в режиме FIFO или THR пуст в режиме без FIFO.

Сигнал RE:

- Становится активным, если RE_EN[0] установлен в 1.
- Становится неактивным, если RE_EN[0] установлен в 0.

Пользователь может выбирать, когда передавать, а когда принимать. И RE и DE могут быть одновременно подтверждены или отменены в любое время. В этом режиме UART не устанавливает время переключения между передачей и приемом или приемом и передачей. Этот режим может использоваться в полнодуплексном режиме работы, где для передачи и приема используется отдельная дифференциальная пара проводов.

34.3.3.2.2 Программно-управляемый полудуплексный режим

Программно-управляемый полудуплексный режим поддерживает одновременную передачу или прием, но не обе одновременно. Переключение между передачей на прием или приемом на передачу осуществляется путем программирования регистров DE_EN и RE_EN.

Сигнал DE:

- Становится активным, если выполняются следующие условия:
    - DE_EN[0] установлен в 1.
    - THR не пуст в режиме без FIFO или FIFO передатчика не пуст в режиме FIFO.

- Если какая-либо передача приема продолжается, то сигнал ожидает, пока приемник не завершит работу и истечения счетчика времени выполнения («RE to DE»).

- Становится неактивным, если выполняются следующие условия:
  - Текущая последовательная передача завершена.
  - DE_EN[0] установлен в 0.
  - Либо FIFO передатчика пуст в режиме FIFO, либо THR пуст в режиме без FIFO.

Сигнал RE:

- Становится активным, если выполняются следующие условия:
  - RE_EN[0] установлен в 1.
  - Если какая-либо передача приема продолжается, то сигнал ожидает окончания передачи и истечения счетчика времени выполнения («DE to RE»).

- Становится неактивным, если выполняются следующие условия:
  - Текущая последовательная передача приема завершена.
  - RE_EN[0] установлен в 0.

Пользователь должен включить либо DE, либо RE, но не оба одновременно. Поскольку сигналы «RE» и «DE» являются взаимоисключающими, пользователь должен убедиться, что они оба не запрограммированы на то, чтобы быть активными в любой момент времени.

34.3.3.2.2.1 Время выполнения от RE до DE.

UART дополняет состоянием ожидания перед переключением из режима приема в режим передачи, как показано на Рисунок 34. 5.

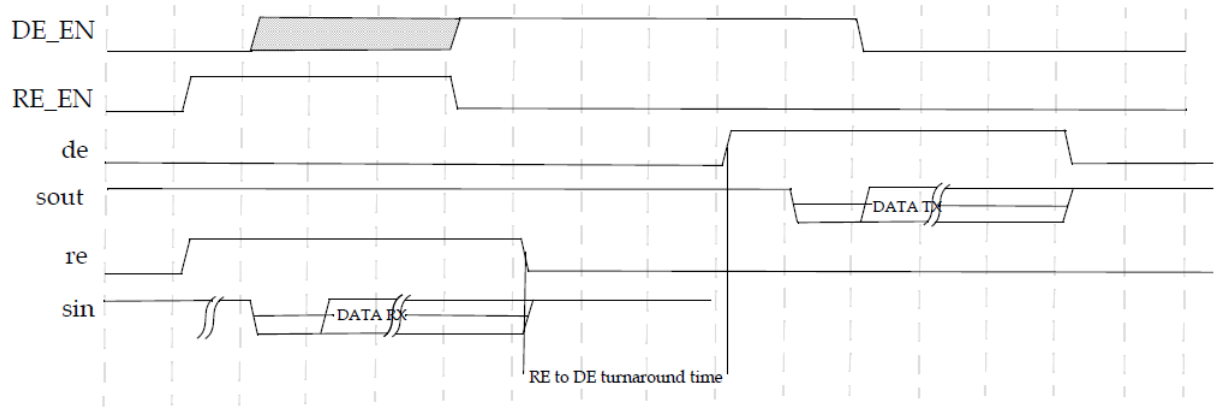


Рисунок 34. 5. Время выполнения от RE до DE

34.3.3.2.2.2 Время выполнения от DE до RE.

UART дополняет состоянием ожидания перед переключением из режима передачи в режим приема, как показано на Рисунок 34. 6.

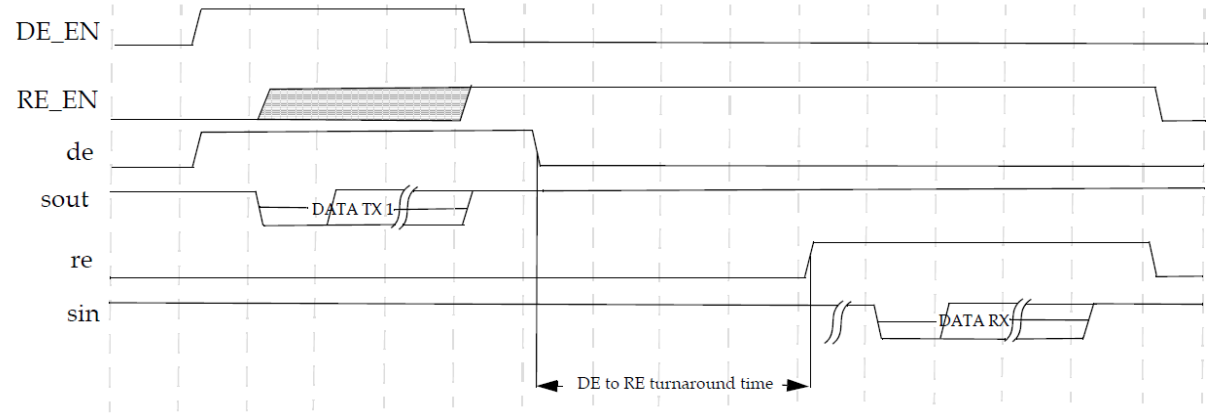


Рисунок 34. 6. Время выполнения от DE до RE

34.3.3.2.3 Полудуплексный режим с аппаратным управлением

Полудуплексный режим с аппаратным управлением поддерживает передачу или прием одновременно, но не оба единовременно. Если оба бита «DE Enable» и «RE Enable» регистров DE_EN и RE_EN включены, переключение между передачей на прием или приемом на передачу происходит автоматически, исходя из состояния Tx-FIFO.

Сигнал DE:

- Становится активным, если выполняются следующие условия:
  - DE_EN[0] установлен в 1.

- THR не пуст в режиме без FIFO или FIFO передатчика не пуст в режиме FIFO.
- Если передача данных на прием продолжается, то сигнал ждет, пока прием не завершится и истечения счетчика времени выполнения («RE to DE»).

- Становится неактивным, если выполняются следующие условия:
  - Текущая последовательная передача завершена.
  - Либо FIFO передатчика пуст в режиме FIFO, либо THR пуст в режиме без FIFO, либо DE_EN[0] установлен в 0.

RE сигнал:

- Становится активным, если выполняются следующие условия:
  - RE_EN[0] установлен в 1.
  - Либо FIFO передатчика пуст в режиме FIFO, либо THR пуст в режиме без FIFO.
  - Если идет передача данных, то сигнал ожидает окончания передачи данных и истечения счетчика времени выполнения («DE to RE»).
- Становится неактивным, если выполняются следующие условия:
  - Текущая последовательная передача приема завершена.
  - Либо FIFO передатчика не пуст в режиме FIFO, либо THR не пуст в режиме без FIFO, либо RE_EN[0] установлен в 0.

34.3.3.3 Примеры сценариев

Рассмотрим сценарий, в котором контролер UART принимает 3 символа и другое устройство UART отправляет эти символы. В то время как 1-й символ принимается контроллером UART, если программное обеспечение выполняет запись в TX FIFO контроллера UART, то в конце первого символа контролер UART переключит режим с приема на передачу. UART отменит утверждение RE и подтвердит сигнал DE. Это приведет к тому, что контроллер UART не будет получать последующие символы.

Следовательно, в полудуплексном режиме с аппаратным переключением пользователь должен убедиться, что данные приема были получены перед записью в Tx-FIFO, чтобы избежать пропуска принимаемых символов.

34.3.3.3.1 Обычный сценарий передачи

Рисунок 34. 7 представляет пример сценария для нормальной передачи.

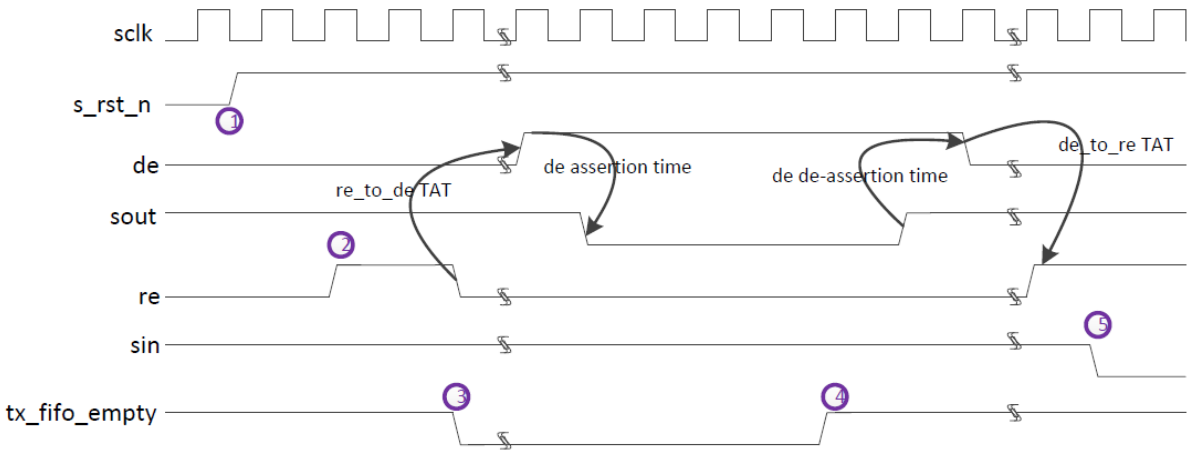


Рисунок 34. 7. Сценарий, где XFER_MODE = 2

На Рисунок 34. 7 показаны следующие действия в различных точках этого сценария:

1. В этот момент сброс снимается, а сигналы DE и RE приводятся к своим сконфигурированным значениям сброса.

2. В этот момент регистры DE_EN и RE_EN устанавливаются 1, и tx_fifo_empty (внутренней сигнал UART) равен 1, что указывает на отсутствие данных в TX FIFO.

3. В этот момент программное обеспечение заполняет TX FIFO, а передача приема не происходит. Поэтому сигнал RE становится низким. Тем не менее, контроллер UART ожидает значения TAT перед подачей сигнала DE. После подачи сигнала DE передача символа начинается с учетом «времени отмены сигнала».

4. В этот момент TX FIFO становится пустым. После передачи текущего символа, UART снимает сигнал DE. Контроллер UART ждет, пока появится значение TAT, прежде чем снова подать сигнал RE.

5. В этот момент контроллер UART начинает получать символ.

34.3.3.3.2 Сценарий приема, когда TX FIFO заполняется

В этом сценарии буфер TX FIFO заполняется при получении символа. В этом случае ожидается, что UART дождется завершения текущего символа, прежде чем сменить роль и начать передачу.

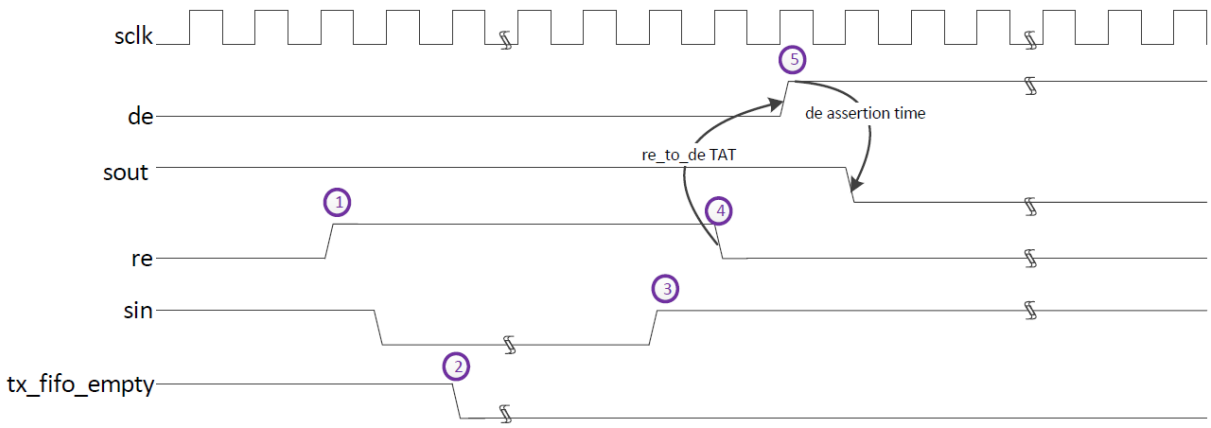


Рисунок 34.8. Прием в процессе, когда TX FIFO заполнен

На Рисунок 34.8 показаны следующие действия в различных моментах этого сценария:

1. Программное обеспечение программирует DE_EN и RE_EN в 1, тем самым подтверждает сигнал RE. После этого контроллер UART начинает получать символ.

2. Tx_fifo_empty понижается. Однако контроллер UART ожидает, пока не будет получен текущий символ, прежде чем утверждать сигнал DE.

3. Входящий символ полностью принят.

4. Сигнал RE становится неактивным после того, как Stop бит полностью принят.

5. После TAT, сигнал DE становится активным, и контроллер UART начинает передачу по истечению времени DET.

34.3.3.3.3 TX FIFO заполнен перед включением регистров DE_EN и RE_EN

В этом случае TX FIFO заполняется до включения DE_EN или RE_EN. Контроллер UART

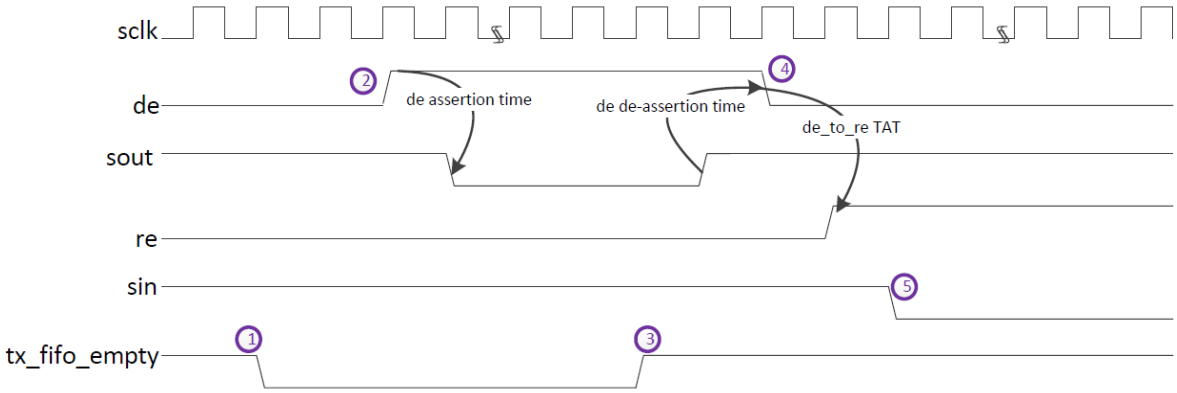


Рисунок 34. 9.TX FIFO запоняется перед включением DE/RE

включает DE вместо RE в этом случае, потому что TX FIFO уже содержит данные для передачи.

На Рисунок 34. 9 показаны следующие действия в различных моментах этого сценария:

1. Tx_fifo_empty устанавливается в 0.
2. DE_EN и RE_EN устанавливается в 1. Поскольку данные уже присутствуют в TX FIFO, контроллер UART подает сигнал DE. UART запускает отправку по истечению времени DET.
3. TX FIFO становится пустым.
4. Сигнал DE отменяется по истечении времени DET. После TAT утверждается сигнал RE.
5. Контроллер UART начинает принимать входящий символ.

34.3.3 Протокол IrDA 1.0 SIR

Режим Infrared Dara Association (IrDA) 1.0 Serial Infrared (SIR) поддерживает двунаправленную передачу данных с удаленными устройствами с использованием инфракрасного излучения в качестве среды передачи. В режиме IrDA 1.0 SIR максимальная скорость передачи данных составляет 115,2 Кбод.

Формат данных аналогичен стандартному формату последовательных данных sout и sin. Каждый символ данных отправляется последовательно в следующем порядке:

1. Начинается с начального бита.
2. За ним следует 8 бит данных.

3. Заканчивается по крайне мере одним Stop битом.

Таким образом, количество передаваемых битов данных фиксировано. Информация о четности не может быть предоставлена, и в этом режиме используется только один Stop бит. Попытка отрегулировать количество отправляемых битов данных или включить четность с помощью регистра LCR не имеет никакого эффекта.

Конфигурация UART для IrDA 1.0 SIR выполняет следующие действия:

- Бит 6 регистра MCR включает или отключает режим IrDA 1.0 SIR.
- Отключение режима IrDA SIR приводит к тому, что логика не реализуется; режим не может быть активирован.
- Когда режим IrDA SIR включен и активен, последовательные данные передаются и принимаются через порты sir_out_n и sir_in.

Передача или отсутствие передачи одного инфракрасного импульса указывает на следующее:

- Передача одного инфракрасного импульса указывает на логический 0.
- Отсутствие передачи импульса указывает на логическую 1.

Длительность каждого импульса составляет 3/16 обычного битового времени. Таким образом, каждый новый символ начинается с инфракрасного импульса в качестве Start бита. Однако полученные данные инвертируются из передаваемых данных из-за инфракрасных импульсов, возбуждающих питание базы фототранзистора приемника IrDA, что снижает его выход. Этот инвертированный транзисторный выход подается на порт UART sir_in, который обеспечивает правильную полярность UART.

На Рисунок 34. 10 показана временная диаграмма для формата данных IrDA SIR по сравнению со стандартным последовательным форматом.

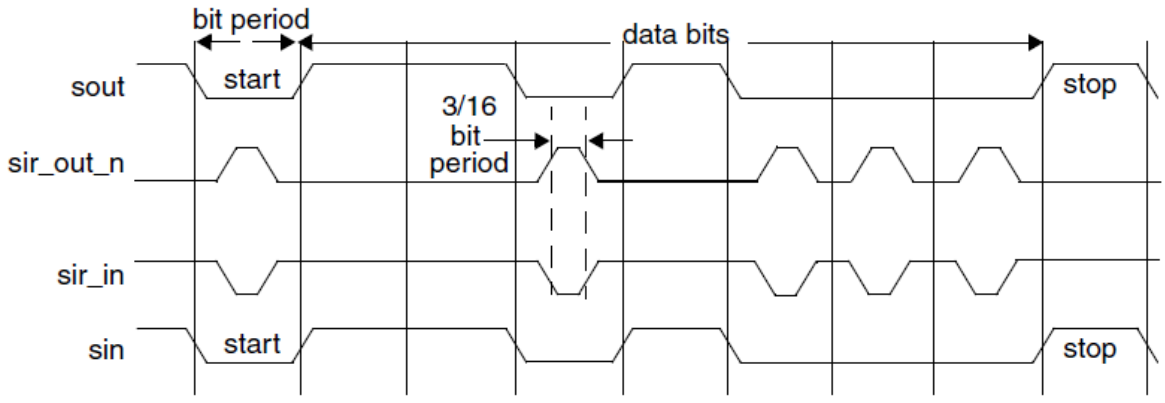


Рисунок 34. 10. Формат данных IrDA SIR

Так как UART может быть сконфигурирован для поддержки режима приема с низким энергопотреблением. Когда UART настроен в этом режиме, можно принимать импульсы SIR длительностью 1,41 микросекунды (минимальная длительность импульса), а также номинальные 3/16 нормального времени последовательного бита. Чтобы использовать этот режим приема с низким энергопотреблением, необходимо запрограммировать регистры LPDLL / LPDLH.

Когда включен режим IrDA SIR, UART работает аналогично отключенному режиму, за одним исключением: передача данных может осуществляться только в полудуплексном режиме, когда включен режим IrDA SIR. Это связано с тем, что физический уровень IrDA SIR определяет минимальную задержку в 10 мс между передачей и приемом; эта задержка в 10 мс должна быть сгенерирована программным обеспечением.

34.3.4 Поддержка FIFO

Можно настроить UART для реализации FIFO, которые буферизируют передачу и прием данных. Если поддержка FIFO не выбрана, то FIFO не реализованы, и только один байт данных приема и байт данных передачи может храниться одновременно в регистрах RBR и THR; это подразумевает 16450-совместимый режим работы. Однако в этом режиме работы большинство расширенных функций недоступны.

В режиме FIFO можно выбрать один из следующих вариантов FIFO:

- FIFO RAM, поставляемые заказчиком.
- Внутренняя RAM на основе D-триггеров (DW_ram_r_w_s_dff).

Если сконфигурированная глубина FIFO превышает 256, выбор памяти FIFO ограничен. Кроме того, выбор внутренней памяти ограничивает тип порта чтения памяти для RAM-портов синхронного чтения на основе D-триггера.

Если выбрана поддержка внешней оперативной памяти, можно использовать синхронную или асинхронную оперативную память. Асинхронная оперативная память обеспечивает считывание данных в течение тактового цикла. Синхронная одноступенчатая оперативная память регистрирует данные по текущему адресу и недоступна до следующего тактового

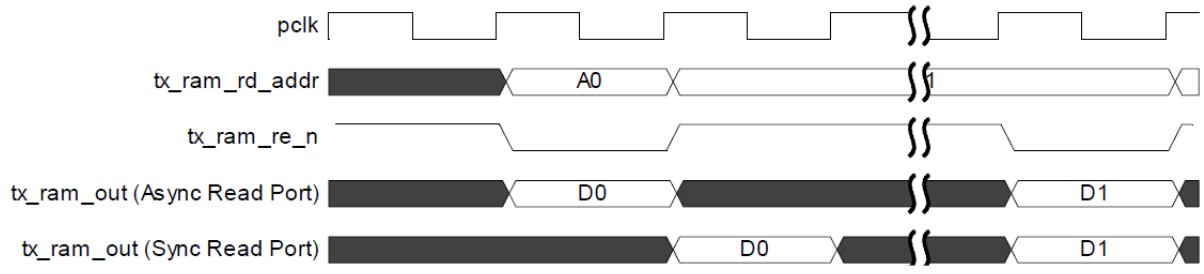


Рисунок 34. 11. Время чтения RAM

цикла. На Рисунок 34. 11 показана временная диаграмма, как для асинхронной, так и для синхронной RAM.

Точно также можно использовать синхронную RAM для записи, которая регистрирует данные по текущему адресу. На Рисунок 34. 12 показана временная диаграмма для записи в RAM.

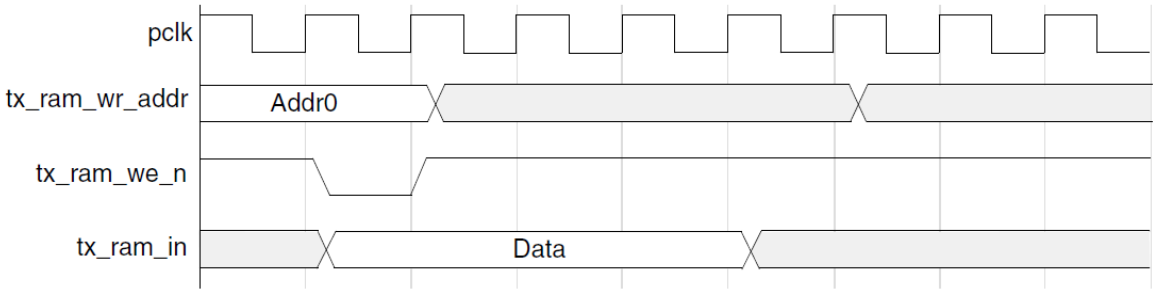


Рисунок 34. 12. Время записи RAM

Когда выбрана поддержка FIFO, для целей тестирования доступен дополнительный программируемый режим доступа FIFO, который позволяет:

- Получить FIFO для записи мастером.
- Передать FIFO для чтения мастером.

Когда режим доступа FIFO не выбран, ни одна из соответствующих логик не реализована, и режим не может быть включен.

Когда выбран режим доступа FIFO, его можно включить с помощью регистра доступа FIFO (FAR [0]). После включения, управляющие части FIFO передачи и приема сбрасываются, и FIFO обрабатываются как пустые.

Данные могут быть записаны в передающие FIFO; однако в этом режиме не происходит никакой последовательной передачи и, таким образом, никакие данные не покидают FIFO. Данные, которые были записаны в FIFO передачи, могут быть считаны обратно с помощью регистра TFR.

Точно так же данные могут быть прочитаны из FIFO приемника. Поскольку в этом режиме работа UART останавливается, данные должны быть записаны в принимающие FIFO, чтобы их можно было прочитать обратно.

Данные записываются в FIFO приемника с использованием регистра RFW. Два старших бита 10-битного регистра используются для записи информации об ошибке кадра и обнаружения ошибок четности в FIFO приемника следующим образом:

- RFW[9] указывает на ошибку кадра.

- RFW[8] указывает на ошибку четности.

Хотя эти биты не могут быть считаны обратно через регистр буфера приема, их можно проверить, прочитав регистр состояния строки и проверив соответствующие биты, когда данные, о которых идет речь, находятся в верхней части FIFO.

34.3.5 Последовательная передача потока данных

Когда FIFO передачи содержит несколько записей данных, UART передает символы из FIFO последовательно. Однако, если параметр конфигурации CLOCK_MODE равен 2, задержки синхронизации UART могут вызвать период бездействия между концом текущего бита Stop и началом следующего бита Start; это проявляется, как увеличенная длительность бита Stop.

34.3.6 Автоматическое управление потоком

UART можно настроить так, чтобы был доступен режим управления последовательным потоком данных. Если FIFO не реализованы, этот режим выбрать невозможно. Когда автоматическое управление потоком не выбрано, никакая соответствующая логика не реализуется, и режим не может быть включен. Когда выбран режим автоматического управления потоком, его можно включить с помощью регистра MCR [5].

На Рисунок 34. 13 показана блок-схема функции автоматического управления потоком.

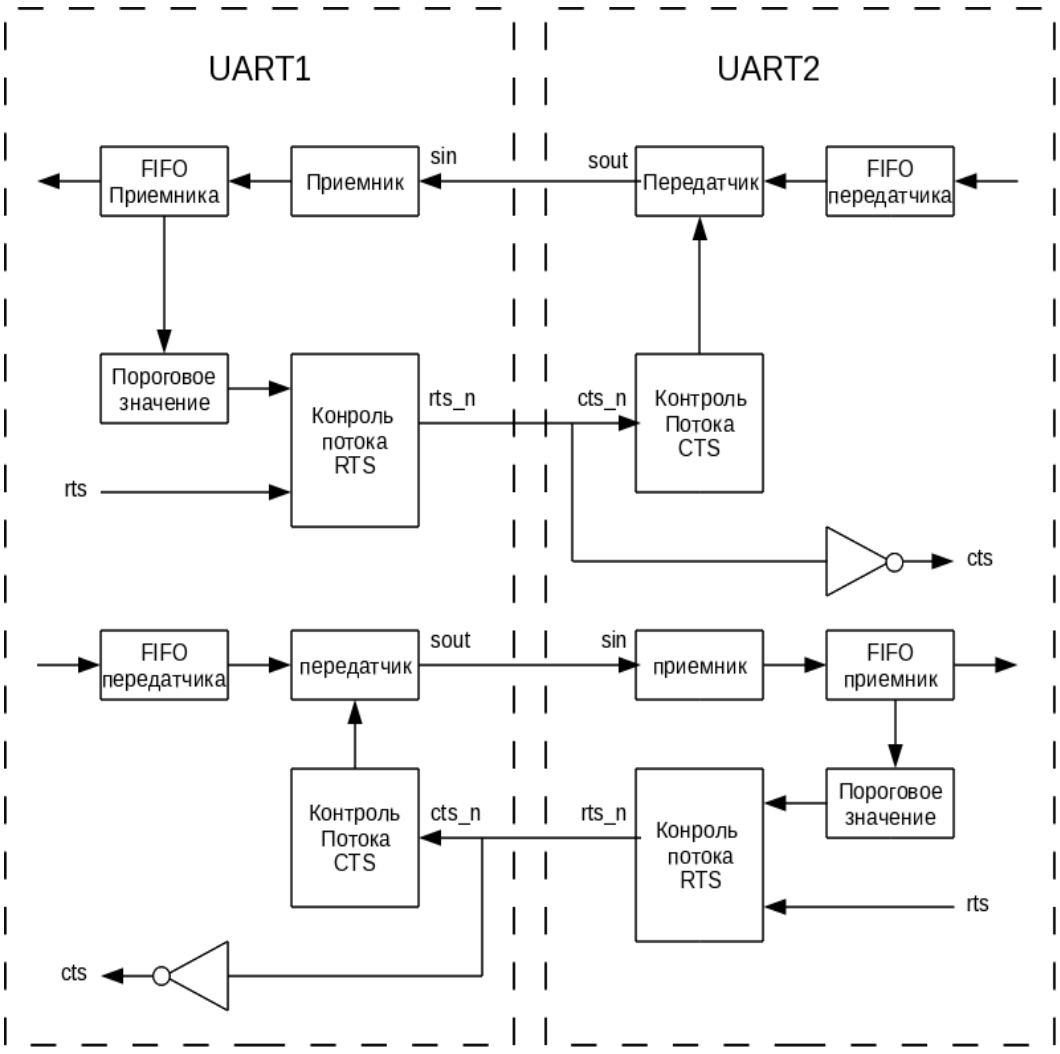


Рисунок 34. 13. Блок-схема автоматического управления потоком

Auto RTS – активируется, когда происходит следующее:

- Автоматическое управление потоком выбирается во время настройки.
- Реализованы FIFO.
- RTS (установлены бит MCR[1] и бит MCR[5]).

- FIFO включены (установлен бит FCR[0]).
- Режим SIR отключен (бит MCR[6] не установлен).

Когда функция Auto RTS включена, выход rts_n принудительно отключается (высокий уровень), когда уровень FIFO приемника достигает порога, установленного FCR[7:6], но только если триггер управления потоком RTC отключен. В противном случае, выход rts_n принудительно неактивен (высокий уровень), когда FIFO почти заполнен, где "почти заполнен" означает два свободных слота в FIFO. Когда rts_n подключен к входу cts_n другого устройства UART, другой UART прекращает отправку последовательных данных, пока в приемнике FIFO не освободится место; то есть, пока он не станет полностью пустым.

Пороговые значения FIFO приемника:

- 1.
- $^1/_4$.
- $^1/_2$.
- 2 меньше, чем полный.

Поскольку один дополнительный символ может быть передан в UART после того, как rts_n стал неактивным - из-за того, что данные уже поступили в блок передатчика в другом UART - установка порога на «2 меньше, чем полный» позволяет максимально использовать FIFO с зоной безопасности в один символ.

Как только FIFO приемника становится полностью пустым в результате чтения RBR, rts_n снова становится активным (низким), сигнализируя другому UART о продолжении отправки данных.

На Рисунок 34. 14 показана временная диаграмма работы Auto RTS.

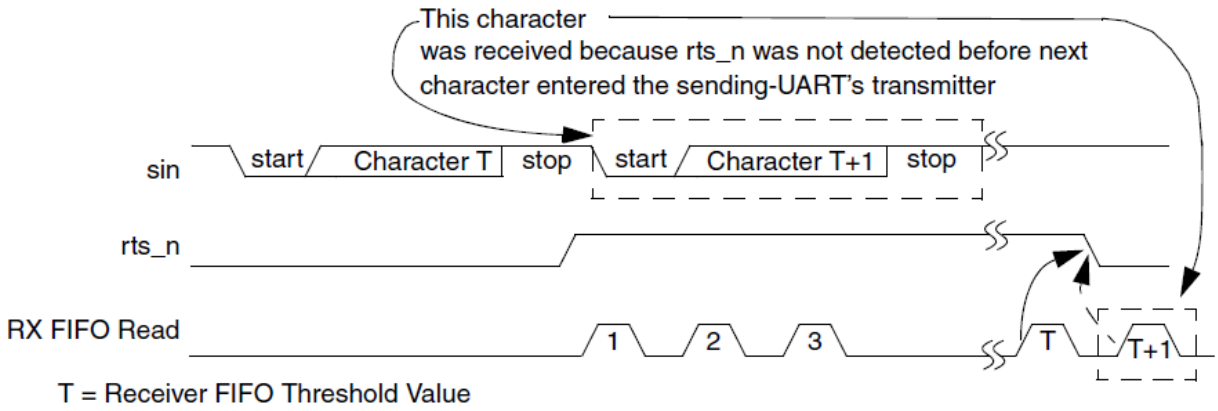


Рисунок 34. 14. Auto RTS

Auto CTS – активируется, когда происходит следующее:

- Автоматическое управление потоком выбирается во время настройки.
- Реализованы FIFO.
- AFCE (MCR[5] = 1).
- FIFO активируются через бит FCR[0] регистра управления FIFO.
- Режим SIR отключен (MCR[6] = 0).

Когда функция Auto CTS включена (активна), передатчик UART отключается всякий раз, когда вход cts_n становится неактивным (высоким); это предотвращает переполнение FIFO принимающего UART.

Если вход cts_n не отключен до середины последнего Stop бита, перед отключением передатчика передается другой символ. В то время как передатчик отключен, в FIFO передатчика все еще можно записать.

Поэтому при использовании этого режима происходит следующее:

- Регистр состояния UART может быть прочитан, чтобы проверить, заполнен ли передающий FIFO (USR[1] установлен в 0).
- Текущий уровень FIFO можно прочитать с помощью регистра TFL.
- Программируемый режим прерывания THRE должен быть включен для доступа к статусу «FIFO full» с помощью LSR.

На Рисунок 34. 15 показана временная диаграмма работы Auto CTS.

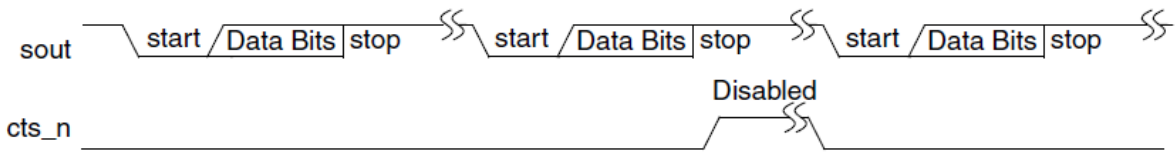


Рисунок 34. 15. Auto CTS

34.3.7 Аппаратное управление потоком данных

Порт UART имеет 2 дополнительных вывода RTS_N и CTS_N, через которые осуществляется автоматическое управление потоком между двумя портами UART. Чтобы включить режим автоматического управления потоком необходимо установить в 1 бит MCR[5].

Выходной сигнал RTS_N переходит в 1 в случае, когда число записей в приемнике FIFO превышает пороговое значение, записанное в битах FCR[6:7]. Когда вывод RTS_N соединен с выводом CTS_N другого порта UART, то другой UART будет вынужден приостановить передачу данных, до тех пор, пока FIFO приемника первого UART не станет пустым. После этого, сигнал RTS_N перейдет в 0 и передача будет продолжена.

34.3.8 Прерывания

UART формирует следующие прерывания:

1. Ошибка приемника.
2. Приемник имеет данные для чтения.
3. Таймаут символа (в режиме FIFO).
4. Регистр передатчика пуст / Буфер передатчика ниже порогового значения (в режиме работы по прерыванию THRE).
5. Контроллер занят.

34.3.9 Программируемое прерывание THRE

UART можно настроить для режима программируемого прерывания THRE, чтобы повысить производительность системы; если FIFO не реализованы, то этот режим не может быть выбран:

- Когда режим программируемого прерывания THRE не выбран, никакая логика не реализуется, и режим не может быть включен.

- Когда выбран режим программируемого прерывания THRE, его можно включить с помощью регистра IER [7].

34.3.10 Включение Clock Gate

UART может быть сконфигурирован так, чтобы иметь выход разрешения синхронизации.

- Когда опция включения clock gate не выбрана, никакая логика не реализуется.
- Когда выбрана опция включения clock gate, сигналы включения - uart_lp_req_pclk для реализаций с одним тактовым сигналом или uart_lp_req_pclk и uart_lp_req_sclk для реализации с двумя тактовыми сигналами - используется для обозначения следующего:
  - Конвейер передачи и приема пуст (нет данных).
  - Никаких действий не произошло.
  - Входные сигналы управления не изменились более чем за один символ – время, необходимое для передачи/приема символа.

Утверждение разрешающих сигналов clock gate указывает на то, что UART неактивен, поэтому тактовые импульсы могут быть гейтированы, чтобы перевести устройство в режим пониженного энергопотребления (LP). Следовательно, следующее должно быть истинным хотя бы для одного символа времени для утверждения сигнала включения clock gate:

- Нет данных в RBR (в режиме без FIFO) или RX FIFO пуст (в режиме FIFO).
- Нет данных в THR (в режиме без FIFO) или TX FIFO пуст (в режиме FIFO).
- sin/sir_in и sout/sir_out_n неактивен (sin/sir_in держится на высоком уровне, sout на высоком уровне или sir_out_n на низком уровне), что указывает на отсутствие активности.
- Нет изменений во входных сигналах управления.

Подтверждение включения clock gate не происходит в следующих режимах работы:

- Режим обратной связи.
- Режим доступа FIFO.
- При передаче паузы.

34.3.11 Поддержка DMA

UART поддерживает DMA с использованием выходных сигналов dma_tx_req_n и dma_rx_req_n, чтобы узнать:

- Когда данные можно читать.
- Когда передающий FIFO пуст.

34.3.11.1 Режимы DMA

UART использует два канала DMA — один для передачи данных и один для приема данных. Существует два режима DMA:

- Режим 0 — 3 бит регистра управления FIFO установлен в 0.
- Режим 1 — 3 бит регистра управления FIFO установлен в 1.

34.3.13.1.1 DMA режим 0

Режим DMA 0 поддерживает единовременную передачу данных DMA.

Сигнал dma_tx_req_n:

- Становится активным-низким при следующих условиях:
  - THR пуст в режиме без FIFO.
  - FIFO передатчика пуст в режиме FIFO с отключенным режимом программируемого прерывания THRE.
  - FIFO передатчика находится на уровне или ниже запрограммированного порога с включенным режимом программируемого прерывания THRE.
- Становится неактивным:
  - Одиночный символ был записан в THR или FIFO передатчика с отключенным режимом программируемого прерывания THRE.
  - FIFO передатчика превышает пороговое значение с включенным режимом программируемого прерывания THRE.

Сигнал dma_rx_req_n:

- Становится активным на низком уровне, когда один символ доступен в FIFO приемника или регистре буфера приема.
- Становится неактивным, когда регистр буфера приема или FIFO приемника пусты, в зависимости от режима FIFO.

34.3.13.1.2 DMA режим 1

Режим DMA 1 поддерживает мульти-DMA передачу данных, при которой несколько передач осуществляются непрерывно до тех пор, пока не опустеет FIFO приемника или не заполнится FIFO передатчика.

Сигнал dma_tx_req_n:

- Когда FIFO передатчика пуст при отключенном режиме программируемого прерывания THRE.
- Когда FIFO передатчика находится на уровне или ниже запрограммированного порога при включенном режиме программируемого прерывания THRE.

Сигнал dma_tx_req_n подается, когда FIFO передатчика полностью заполнен.

Сигнал dma_rx_req_n:

- Когда FIFO приемника находится на уровне или выше запрограммированного уровня срабатывания.
- Когда истекло время ожидания символа; ERBFI не нужно устанавливать.

Сигнал dma_rx_req_n отменяется, когда FIFO приемника становится пустым.

34.3.12 Аппаратный интерфейс запросов PDMA

Контроллеры UART0, UART1, UART2 имеют аппаратный интерфейс запросов PDMA. Запросы формируются следующим образом.

Запрос к PDMA на запись в UART формируется:

1. При выключенном FIFO, когда регистр THR пуст.
2. При включенном FIFO но выключенном прерывании THRE, когда FIFO передатчика пуст.
3. При включенном FIFO и включенном прерывании THRE, когда кол-во слов данных в FIFO передатчика ниже установленного порогового значения, установленного в поле TET регистра FCR.

Запрос к PDMA на чтение из UART формируется:

1. При выключенном FIFO, когда в регистре RBR есть данные.
2. При включенном FIFO, когда кол-во слов данных в FIFO приемника равно или превышает пороговое значение, установленное в поле RFTL регистра FCR.

Размер всех пересылок UART-PDMA контролирует только PDMA.

34.3.13 Сигнал сброса

При настройке для асинхронной последовательной синхронизации UART включает два отдельных сигнала сброса, каждый из которых предназначен для своего собственного домена синхронизации:

- Present сбрасывает логику в pclk.
- s_rst_n сбрасывает логику с sclk.

Во избежание серьезных сбоев в работе оба тактовых домена UART должны быть сброшены до того, как будет предпринята любая попытка отправить или получить данные по последовательной линии; то есть сбросить только один тактовый домен UART без сброса другого тактового домена является недопустимой.

Каждый сигнал сброса должен быть отключен синхронно с соответствующим тактовым сигналом.

При утверждении сигналов сброса сигнал s_rst_n должен быть заявлен до или в то же время, что и presetn; это предотвращает любую неожиданную активность на последовательной линии, которая может возникнуть в результате сброса регистров программирования без сброса последовательной логики.

Аналогично, при отмене подтверждения сигналов сброса, s_rst_n должен быть отменен до отмены presetn. Самая безопасная процедура сброса UART следующая:

- Утвердить s_rst_n и presetn; последовательность утверждения этих двух сигналов и их временные отношения с sclk и pclk не важны.
- Снять сигнал s_rst_n синхронно с sclk.
- Снять сигнал present синхронно с pclk.

Оба сигнала сброса должны быть активны как минимум в течение трех циклов соответствующего тактового сигнала.

35. КОНТРОЛЛЕР ШИМ (PWM)

35.1 Общая информация

Широтно-импульсный модулятор PWM (Pulse Width Modulator) генерирует выходные импульсы сложной формы при минимальном участии CPU. Блок отличается высокой гибкостью работы и простотой программирования.

Блок PWM состоит из 2-х каналов PWM_CH.

Каждый канал PWM_CH имеет следующие особенности:

- 32-битный специализированный счетчик с возможностью контроля периода, частоты и направления счета;
- два независимых выхода сформированных сигналов OUTA и OUTB;
- асинхронный программный контроль выходных сигналов;
- программное управление сдвигом фазы выходных сигналов относительно фазы выходных сигналов других каналов PWM;
- синхронизация относительно других каналов во время пуска и во время работы;
- возможность генерация запретной зоны от различных фронтов с различным периодом;
- контроль выходных сигналов по внешним входам TU;
- возможность наложения высокочастотного сигнала на выходы OUTA и OUTB;
- управление формированием запроса на прерывание CPU.

35.2 Регистры

Базовый адрес индивидуальных регистров каналов PWM:

- PWM_CH0_OFFSET = 0x0177_0000;
- PWM_CH1_OFFSET = 0x0177_0100;

Базовый адрес индивидуальных регистров каналов PWM:

- PWM_GLOBAL_OFFSET = 0x0177_0X00;

Адрес индивидуального регистра вычисляется по формуле:

ADDR_REG = PWM_CH*_OFFSET + REG_OFFSET

Адрес глобального регистра вычисляется по формуле:

ADDR_REG = PWM_GLOBAL_OFFSET + REG_OFFSET

Перечень регистров приведен в Таблица 35.1.

Таблица 35.1. Перечень программно-доступных регистров

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
CTRRUN*	Регистр управления пуском/остановом таймера	0x00000000	R/W	0x80
CTRSTS*	Регистр статуса счетчика	0x01010101	R/W	0x84
CLKCTL	Регистр управления параметрами блока Count unit	0x0000003A	R/W	0x08
CTRPHS	Регистр фазы синхронизации блока Count unit	0x00000000	R/W	0x0C
CTRPRD	Регистр периода счета блока Count unit	0x00000000	R/W	0x10
CTRCNT	Регистр счетчика блока Count unit	0x00000000	R/W	0x14
CMPCTL	Регистр управления параметрами блока сравнения Compare unit	0x00000000	R/W	0x20
CMPA	Регистр сравнения CMPA блока сравнения Compare unit	0x00000000	R/W	0x24
CMPB	Регистр сравнения CMPB блока сравнения Compare unit	0x00000000	R/W	0x28
EMCTLA	Управляющий регистр выхода OUTA блока Event manager	0x00000000	R/W	0x2C
EMCTLB	Управляющий регистр выхода OUTB блока Event manager	0x00000000	R/W	0x30
EMSWFR	Регистр программного управления выходами OUTA/OUTB с однократным действием блока Event manager	0x00000000	R/W	0x34
EMCSWFR	Регистр программного управления выходами OUTA/OUTB с продолжительным действием блока Event manager	0x00000000	R/W	0x38
DZCTL	Регистр управления блока Dead zone	0x00000000	R/W	0x40
DZRPER	Регистр периода задержки после переднего фронта блока Dead zone	0x00000000	R/W	0x44
DXFPER	Регистр периода задержки после заднего фронта блока Dead zone	0x00000000	R/W	0x48
CHCTL	Регистр управления блока Chopper	0x00000000	R/W	0x4C
TUSEL	Регистр выбора принципа работы блока Trip unit	0x00000000	R/W	0x50
TUCTL	Регистр определения реакции на событие блока Trip unit	0x00000000	R/W	0x54
TUINTM	Регистр маски прерывания блока Trip unit	0x00000000	R/W	0x58
TUSTS	Регистр статуса блока Trip unit	0x00000000	R	0x5C
TUCLR	Регистр очистки битов статуса блока Trip unit	0x00000000	R/W	0x60
TUFRC	Регистр программного формирования событий блока Trip unit	0x00000000	R/W	0x64
ICSEL	Регистр выбора принципа работы блока Interrupt control	0x00000000	R/W	0x68
ICCTL	Регистр определения реакции на событие блока Interrupt control	0x00000000	R/W	0x6C
ICSTS	Регистр статуса блока Interrupt control	0x00000000	R	0x70
ICCLR	Регистр очистки битов статуса блока Interrupt control	0x00000000	R/W	0x74
ICFRC	Регистр программного формирования событий блока Interrupt control	0x00000000	R/W	0x78

“*” помечены глобальные регистры

35.2.1 Глобальные регистры

Особенностью глобальных регистров является то, что с их помощью осуществляется управление одновременно всеми каналами PWM.

35.2.1.1 Регистр управления пуском/остановом таймеров CRTRUN

Регистр CTRRUN (Таблица 35.2) используется для управления пуском/остановом таймера и режимом работы предделителя частоты всех каналов.

Таблица 35.2. Регистр CTRRUN

Номер разряда	Условное обозначение	Описание	Доступ	Исходное состояние
[31:14]	–	Резерв		
[13]	PRESMODE1	Управление режимом работы предделителя канала PWM_CH1 0 – предделитель формирует частоту только при включенном таймере (RUN1==2'b1x) 1 – предделитель формирует частоту не зависимо от RUN1. Генерация тактового сигнала начинается сразу после записи «1»	R/W	0
[12]	PRESRST1	Управление состоянием предделителя канала PWM_CH1 в момент простоя (PRESMODE1==0 и RUNSTS1==1'b0) 0 – счетчик предделителя сбрасывается в «0» 1 – счетчик предделителя сохраняет состояние на момент останова	R/W	0
[11:10]	–	Резерв		
[9:8]	RUN1	Управление пуском/остановкой канала PWM_CH1 00b – остановка после следующего переключения счетчика CTRCNT 01b – остановка при совершении следующих событий: up-count режим: остановка при CTRCNT==CTRPRD down-count режим: остановка при CTRCNT==0 up-down-count режим: остановка при CTRCNT==0 1xb – пуск	R/W	0
[7:6]	–	Резерв		
[5]	PRESMODE0	Управление режимом работы предделителя канала PWM_CH0 0 – предделитель формирует частоту только при включенном таймере (RUN0==2'b1x) 1 – предделитель формирует частоту не зависимо от RUN0. Генерация тактового сигнала начинается сразу после записи «1»	R/W	0
[4]	PRESRST0	Управление состоянием предделителя канала PWM_CH0 в момент простоя (PRESMODE0==0 и RUNSTS0==1'b0) 0 – счетчик предделителя сбрасывается в «0» 1 – счетчик предделителя сохраняет состояние на момент останова	R/W	0
[3:2]	–	Резерв		
[1:0]	RUN0	Управление пуском/остановкой канала PWM_CH0	R/W	0

Номер разряда	Условное обозначение	Описание	Доступ	Исходное состояние
		00b – остановка после следующего переключения счетчика CTRCNT 01b – остановка при совершении следующих событий: up-count режим: остановка при CTRCNT==CTRPRD down-count режим: остановка при CTRCNT==0 up-down-count режим: остановка при CTRCNT==0 1xb – пуск		

35.2.1.2 Регистр статуса счетчиков CTRSTS

Регистр CTRSTS (Таблица 35.3) отображает текущее состояние подсистемы Count.

Таблица 35.3. Регистр CTRSTS

Номер разряда	Условное обозначение	Описание	Доступ	Исходное состояние
[31:11]	–	Резерв		
[10]	RUN_STS1	Бит, отображающий состояние канала PWM_CH1 0 – счетчик остановлен 1 – счетчик работает	R	0
[9]	SYNC_STS1	Бит статуса входа синхронизации SYNCI канала PWM_CH1 0 – запись нуля не имеет эффекта, чтение нуля говорит об отсутствии внешней синхронизации 1 – чтение единицы говорит о том, что произошла синхронизация, запись единицы очищает данный бит	R/W	0
[8]	DIR_STS1	Отображает текущее направление счета CTRCNT канала PWM_CH1. Этот бит не несет никакой информации, пока таймер не запущен. 0 – счет осуществляется вниз 1 – счет осуществляется вверх	R	1
[7:3]	–	Резерв		
[2]	RUN_STS0	Бит отображающий состояние канала PWM_CH0 0 – счетчик остановлен 1 – счетчик работает	R	0
[1]	SYNC_STS0	Бит статуса входа синхронизации SYNCI канала PWM_CH0 0 – запись нуля не имеет эффекта, чтение нуля говорит об отсутствии внешней синхронизации 1 – чтение единицы говорит о том, что произошла синхронизация, запись единицы очищает данный бит	R/W	0
[0]	DIR_STS0	Отображает текущее направление счета CTRCNT канала PWM_CH0. Этот бит не несет никакой информации, пока таймер не запущен. 0 – счет осуществляется вниз 1 – счет осуществляется вверх	R	1

35.2.2 Регистры блока Count unit

35.2.2.1 Регистр управления подсистемой синхронизации CLKCTL (Таблица 35.4)

Таблица 35.4. Регистр CLKCTL

Номер разряда	Условное обозначение	Описание	Доступ	Исходное состояние
[31:20]	–	Резерв		
[19:17]	DIVMUX	Управление мультиплексором делителя частоты 2: 0h – частота от предделителя делится на 1; 1h – частота от предделителя делится на 2; 2h – частота от предделителя делится на 4; 3h – частота от предделителя делится на 8; 4h – частота от предделителя делится на 16; 5h – используется внешняя частота PWM_CLK **Важно!** Менять значение возможно только при выключенном предделители CLKCTL[PRESCEN] == 0	R/W	0
[16]	SYNCRST	Бит разрешения сброса предделителя при возникновении событий SYNCI или SWFSYNC 0 – сброс запрещен 1 – сброс разрешен	R/W	0
[15:8]	PRESPRD	Значение предделителя частоты CLK	R/W	0
[7]	DIRSYNC	Направление счета после синхронизации. Этот бит используется, только когда счетчик работает в up-down режиме. 0 – после синхронизации счетчик декрементируется; 1 – после синхронизации счетчик инкрементируется	R/W	0
[6]	SWFSYNC	Программный импульс синхронизации. 0 – запись нуля не дает никакого эффекта, при чтении всегда возвращается 0; 1 – запись 1 провоцирует формирование импульса синхронизации, который по ИЛИ соединяется со входом SYNCI. SWFSYNC	R/W	0
[5:4]	SYNCOSEL	Выбор источника выходного сигнала SYNCO: 0h – SYNCI; 1h – счетчик равен нулю (CTRCNT==0); 2h – счетчик равен значению регистра сравнения B (CTRCNT==CMPB); 3h – SYNCO отключен	R/W	3h
[3]	LOADPRD	Управление моментом переписи данных из теневого регистра периода в активный 0 – регистр периода (CTRPRD) загружается из теневого регистра, когда счетчик (CTRCNT) равен нулю (запись или чтение CTRPRD осуществляется через теневой регистр); 1 – CTRPRD загружается напрямую без использования теневого регистра (запись или чтение CTRPRD осуществляется напрямую).	R/W	1h
[2]	SYNCPHSEN	Сигнал разрешения загрузки счетчика из регистра фазы 0 – загрузка CTRCNT из регистра фазы CTRPHS запрещена	R/W	0

Номер разряда	Условное обозначение	Описание	Доступ	Исходное состояние
		1 – загрузка CTRCNT из регистра фазы CTRPHS во время синхронизации разрешена		
[1:0]	CNTMODE	Режим работы счетчика CRTCNT. 0h – up-count режим; 1h – down-count режим; 2h – up-down-count режим; 3h – счет не осуществляется	R/W	2h

35.2.2.2 Регистр фазы синхронизации CTRPHS (Таблица 35.5)

Таблица 35.5. Регистр CTRPHS

Номер разряда	Условное обозначение	Описание	Доступ	Исходное состояние
[31:0]	CTRPHS	Содержит значение, в которое переключится счетчик CTRCNT во время синхронизации. Если CLKCTL[SYNCPHSEN]=0, то при возникновении события синхронизации счетчик CTRCNT не переходит в значение, прописанное в регистре CTRPHS. Если CLKCTL[SYNCPHSEN]=1, то при возникновении события синхронизации счетчик CTRCNT загружает значение регистра CTRPHS.	R/W	0

35.2.2.3 Регистр счетчика CTRCNT (Таблица 35.6)

Таблица 35.6. Регистр CTRCNT

Номер разряда	Условное обозначение	Описание	Доступ	Исходное состояние
[31:0]	CTRCNT	При чтении данный регистр отображает текущее значение счетчика CTRCNT. При записи прописывается новое значение счетчика CTRCNT без синхронизации с частотой работы счетчика и без использования теневого регистра.	R/W	0

35.2.2.4 Регистр периода счета CTRPRD (Таблица 35.7)

Таблица 35.7. Регистр CTRPRD

Номер разряда	Условное обозначение	Описание	Доступ	Исходное состояние
[31:0]	CTRPRD	Данное поле определяет период счетчика CTRCNT и, соответственно, устанавливает частоту PWM. Теневой регистр включается и выключается битом CLKCTL[LOADPRD]. Если CLKCTL[LOADPRD]=0, теневой регистр включен, и любая запись или чтение проходят через него. Активный регистр загружается из теневого в момент, когда CTRCNT=0 Если CLKCTL[LOADPRD]=1, теневой регистр выключен, и любая запись или чтение проходит напрямую без участия теневого регистра.	R/W	0

Номер разряда	Условное обозначение	Описание	Доступ	Исходное состояние
		Активный и теневой регистры имеют единый адрес.		

35.2.3 Регистры блока Compare unit.

35.2.3.1 Регистр управления параметрами блока сравнения Compare unit CMPCTL (Таблица 35.8)

Таблица 35.8. Регистр CMPCTL

Номер разряда	Условное обозначение	Описание	Доступ	Исходное состояние
[31:8]	-			
[7]	SCMPBFULL	Признак наличия данных в теневом регистре сравнения CMP*B* не загруженных в активный регистр сравнения CMP*B*. Бит сбрасывается во время загрузки данных в активный регистр. 0 – нет не загруженных данных 1 – есть не загруженные данные	R	0
[6]	SCMPAFULL	Признак наличия данных в теневом регистре сравнения CMP*A* не загруженных в активный регистр сравнения CMP*A*. Бит сбрасывается во время загрузки данных в активный регистр. 0 – нет не загруженных данных 1 – есть не загруженные данные	R	0
[5]	SCMPBMODE	Режим работы регистра *CMPB* 0 – работа с теневым регистром, все запросы *CPU* проходят через теневой регистр; 1 – прямой режим, используется только активный регистр *CMPB*	R/W	0
[4]	SCMPAMODE	Режим работы регистра CMPA 0 – работа с теневым регистром, все запросы CPU проходят через теневой регистр; 1 – прямой режим, используется только активный регистр CMPA	R/W	0
[3:2]	LDBMODE	Выбор режима загрузки данных из теневого регистра в активный CMPB. Данное поле не имеет значения при работе в режиме прямой загрузки CMPCTL[SCMPBMODE]=1 0h – загрузка при CTRCNT=0 1h – загрузка при CTRCNT=CTRPRD 2h – загрузка при CTRCNT=0 или CTRCNT=CTRPRD 3h – загрузка не осуществляется	R/W	0
[1:0]	LDAMODE	Выбор режима загрузки данных из теневого регистра в активный CMPA. Данное поле не имеет значения при работе в режиме прямой загрузки CMPCTL[SCMPAMODE]=1 0h – загрузка при CTRCNT=0 1h – загрузка при CTRCNT=CTRPRD 2h – загрузка при CTRCNT=0 или CTRCNT=CTRPRD 3h – загрузка не осуществляется	R/W	0

35.2.3.2 Регистр сравнения CMPA (Таблица 35.9)

Таблица 35.9. Регистр CMPA

Номер разряда	Условное обозначение	Описание	Доступ	Исходное состояние
[31:0]	CMPA	Содержимое активного регистра CMPA сравнивается со значением счетчика CRTCNT. Если значения равны, то блок сравнения генерирует событие «CMPA==CTRCNT». По этому событию менеджер событий (Event manager) совершает одно или более действие. Эти действия могут быть направлены на формирование выходных сигналов OUTA или OUTB в зависимости от конфигурации регистров EMCTLA и EMCTLB. Реакция на событие может быть следующая: не производить никаких действий; зажать OUTA и/или OUTB в 0; зажать OUTA и/или OUTB в 1; инвертировать OUTA и/или OUTB Теневой регистр включен по умолчанию и управляется битом CMPCTL[SCMPAMODE] Если CMPCTL[SCMPAMODE]=0, то теневой регистр включен, запись/чтение проходит через него. В этом случае CMPCTL[LDAMODE] определяет момент загрузки данных в активный регистр. Перед записью необходимо удостовериться, что в теневом регистре нет не переданных в активный регистр данных проанализировав CMPCTL[SCMPAFULL] бит. Если CMPCTL[SCMPAMODE]=1, то теневой регистр не используется, запись/чтение осуществляется с активным регистром. В любом режиме работы активный и теневой регистры имеют один адрес.	R/W	0

35.2.3.3 Регистр сравнения CMPB (Таблица 35.10)

Таблица 35.10. Регистр CMPB

Номер разряда	Условное обозначение	Описание	Доступ	Исходное состояние
[31:0]	CMPB	Содержимое активного регистра CMPB сравнивается со значением счетчика CRTCNT. Если значения равны, то блок сравнения генерирует событие «CMPB==CTRCNT». По этому событию менеджер событий (Event manager) совершает одно или более действие. Эти действия могут быть направлены на формирование выходных сигналов OUTA или OUTB в зависимости от конфигурации регистров EMCTLA и EMCTLB. Реакция на событие может быть следующая: не производить никаких действий; зажать OUTA и/или OUTB в 0; зажать OUTA и/или OUTB в 1; инвертировать OUTA и/или OUTB	R/W	0

Номер разряда	Условное обозначение	Описание	Доступ	Исходное состояние
		Теневой регистр включен по умолчанию и управляется битом CMPCTL[SCMPBMODE] Если CMPCTL[SCMPBMODE]=0, то теневой регистр включен, запись/чтение проходит через него. В этом случае CMPCTL[LDBMODE] определяет момент загрузки данных в активный регистр. Перед записью необходимо удостовериться, что в теневом регистре нет не переданных в активный регистр данных проанализировав CMPCTL[SCMPBFULL] бит. Если CMPCTL[SCMPBMODE]=1, то теневой регистр не используется, запись/чтение осуществляется с активным регистром. В любом режиме работы активный и теневой регистры имеют один адрес.		

35.2.4 Регистры блока Event manager

35.2.4.1 Регистр управления выхода OUTA блока Event manager EMCTLA (Таблица 35.11)

Таблица 35.11. Регистр EMCTLA

Номер разряда	Условное обозначение	Описание	Доступ	Исходное состояние
[31:12]	–	Резерв		
[11:10]	ECMPBD	Действие, которое нужно выполнить при декременте счетчика и CTRCNT=CMPB 0h – не выполнять действий; 1h – clear: сбросить OUTA в 0 2h – set: установить OUTA в 1 3h – toggle: инвертировать OUTA	R/W	0
[9:8]	ECMPBI	Действие, которое нужно выполнить при инкременте счетчика и CTRCNT=CMPB 0h – не выполнять действий; 1h – clear: сбросить OUTA в 0 2h – set: установить OUTA в 1 3h – toggle: инвертировать OUTA	R/W	0
[7:6]	ECMPAD	Действие, которое нужно выполнить при декременте счетчика и CTRCNT=CMPA 0h – не выполнять действий; 1h – clear: сбросить OUTA в 0 2h – set: установить OUTA в 1 3h – toggle: инвертировать OUTA	R/W	0
[5:4]	ECMPAI	Действие, которое нужно выполнить при инкременте счетчика и CTRCNT=CMPA 0h – не выполнять действий; 1h – clear: сбросить OUTA в 0 2h – set: установить OUTA в 1 3h – toggle: инвертировать OUTA	R/W	0
[3:2]	EPRD	Действие, которое нужно выполнить при CTRCNT=CTRPRD 0h – не выполнять действий; 1h – clear: сбросить OUTA в 0 2h – set: установить OUTA в 1	R/W	0

Номер разряда	Условное обозначение	Описание	Доступ	Исходное состояние
		3h – toggle: инвертировать OUTA		
[1:0]	EZRO	Действие, которое нужно выполнить при CTRCNT=0 0h – не выполнять действий; 1h – clear: сбросить OUTA в 0 2h – set: установить OUTA в 1 3h – toggle: инвертировать OUTA	R/W	0

35.2.4.2 Регистр управления выхода OUTB блока Event manager EMCTLB (Таблица 35.12)

Таблица 35.12. Регистр EMCTLB

Номер разряда	Условное обозначение	Описание	Доступ	Исходное состояние
[31:12]	–	Резерв		
[11:10]	ECMPBD	Действие, которое нужно выполнить при декременте счетчика и CTRCNT=CMPB 0h – не выполнять действий; 1h – clear: сбросить OUTB в 0 2h – set: установить OUTB в 1 3h – toggle: инвертировать OUTB	R/W	0
[9:8]	ECMPBI	Действие, которое нужно выполнить при инкременте счетчика и CTRCNT=CMPB 0h – не выполнять действий; 1h – clear: сбросить OUTB в 0 2h – set: установить OUTB в 1 3h – toggle: инвертировать OUTB	R/W	0
[7:6]	ECMPAD	Действие, которое нужно выполнить при декременте счетчика и CTRCNT=CMPA 0h – не выполнять действий; 1h – clear: сбросить OUTB в 0 2h – set: установить OUTB в 1 3h – toggle: инвертировать OUTB	R/W	0
[5:4]	ECMPAI	Действие, которое нужно выполнить при инкременте счетчика и CTRCNT=CMPA 0h – не выполнять действий; 1h – clear: сбросить OUTB в 0 2h – set: установить OUTB в 1 3h – toggle: инвертировать OUTB	R/W	0
[3:2]	EPRD	Действие, которое нужно выполнить при CTRCNT=CTRPRD 0h – не выполнять действий; 1h – clear: сбросить OUTB в 0 2h – set: установить OUTB в 1 3h – toggle: инвертировать OUTB	R/W	0
[1:0]	EZRO	Действие, которое нужно выполнить при CTRCNT=0 0h – не выполнять действий; 1h – clear: сбросить OUTB в 0 2h – set: установить OUTB в 1 3h – toggle: инвертировать OUTB	R/W	0

35.2.4.3 Регистр программного управления выходами OUTA/OUTB с однократным действием блока Event manager EMSWFR (Таблица 35.13)

Таблица 35.13. Регистр EMSWFR

Номер разряда	Условное обозначение	Описание	Доступ	Исходное состояние
[31:8]	–	Резерв		
[7:6]	LDCSWRF	Определяет механизм загрузки активного регистра EMCSWFR из теневого регистра 0h – загрузка при CTRCNT=0 1h – загрузка при CTRCNT=CTRPRD 2h – загрузка при CTRCNT=0 или CTRCNT=CTRPRD 3h – загружать напрямую при обращении CPU без использования теневого регистра.	R/W	0
[5]	ONESFB	Программный вызов единичного переключения для выхода OUTB 0 – запись 0 не вызывает эффекта, читается всегда 0 1 – запись 1 генерирует однократное переключение выхода OUTB, затем данный бит сбрасывается в 0	R/W	0
[4:3]	ACTSFB	Действие, которое нужно выполнить при программном вызове переключения OUTB 0h – не выполнять действий; 1h – clear: сбросить OUTB в 0 2h – set: установить OUTB в 1 3h – toggle: инвертировать OUTB	R/W	0
[2]	ONESFA	Программный вызов единичного переключения для выхода OUTA 0 – запись 0 не вызывает эффекта, читается всегда 0 1 – запись 1 генерирует однократное переключение выхода OUTA, затем данный бит сбрасывается в 0	R/W	0
[1:0]	ACTSFA	Действие, которое нужно выполнить при программном вызове переключения OUTA 0h – не выполнять действий; 1h – clear: сбросить OUTA в 0 2h – set: установить OUTA в 1 3h – toggle: инвертировать OUTA	R/W	0

35.2.4.4 Регистр программного управления выходами OUTA/OUTB с продолжительным действием блока Event manager EMCSWFR (Таблица 35.14)

Таблица 35.14. Регистр EMCSWFR

Номер разряда	Условное обозначение	Описание	Доступ	Исходное состояние
[31:4]	–	Резерв		
[3:2]	LONGSFB	Продолжительное программное воздействие на OUTB При использовании теневого регистра эффект от воздействия продолжительного зажима выхода наступает на следующий такт CTRCNT после загрузки данных в активный регистр	R/W	0

		Во время работы без теневого регистра эффект наступает на следующий такт CTRCNT Управление режимом работы осуществляется при помощи бита EMSWFR[LDCSWRF] 0h – нет воздействия 1h – OUTB продолжительно зажимается в 0 2h – OUTB продолжительно зажимается в 1 3h – программное воздействие запрещено		
[1:0]	LONGSFA	Продолжительное программное воздействие на OUTA При использовании теневого регистра эффект от воздействия продолжительного зажима выхода наступает на следующий такт CTRCNT после загрузки данных в активный регистр Во время работы без теневого регистра эффект наступает на следующий такт CTRCNT Управление режимом работы осуществляется при помощи бита EMSWFR[LDCSWRF] 0h – нет воздействия 1h – OUTA продолжительно зажимается в 0 2h – OUTA продолжительно зажимается в 1 3h – программное воздействие запрещено	R/W	0

35.2.5 Регистры блока Dead zone.

35.2.5.1 Регистр управления блока Dead zone DZCTL (Таблица 35.15)

Таблица 35.15. Регистр DZCTL

Номер разряда	Условное обозначение	Описание	Доступ	Исходное состояние
[31:4]	–	Резерв		
[5]	INMUXF	Выбор источника сигнала для генерации запрещенной зоны после заднего фронта 0 – подключен сигнал OUTA in 1 – подключен сигнал OUTB in	R/W	0
[4]	INMUXR	Выбор источника сигнала для генерации запрещенной зоны после переднего фронта 0 – подключен сигнал OUTA in 1 – подключен сигнал OUTB in	R/W	0
[3]	INVMUXB	Управление полярностью OUTB после генерации запрещенной зоны 0 – нет инверсии 1 – есть инверсия	R/W	0
[2]	INVMUXA	Управление полярностью OUTA после генерации запрещенной зоны 0 – нет инверсии 1 – есть инверсия	R/W	0
[1]	OUTMUXB	Выбор режима работы блока dead zone при формирования OUTB 0 – генератор запрещенной зоны после заднего фронта выключен, OUTB транслируется без изменений, поля регистра DZCTL[INMUXF] и DZCTL[INVMUXB] ни на что не влияют 1 - генератор запрещенной зоны после заднего фронта включен, входной сигнал для формирования OUTB определяется битом DZCTL[INMUXF]	R/W	0

Номер разряда	Условное обозначение	Описание	Доступ	Исходное состояние
[0]	OUTMUXA	Выбор режима работы блока dead zone при формировании OUTA 0 – генератор запрещенной зоны после переднего фронта выключен, OUTA транслируется без изменений, поля регистра DZCTL[INMUXR] и DZCTL[INVMUXA] ни на что не влияют 1 - генератор запрещенной зоны после заднего фронта включен, входной сигнал для формирования OUTA определяется битом DZCTL[INMUXR]	R/W	0

35.2.5.2 Регистр периода задержки после переднего фронта блока Dead zone DZRPER (Таблица 35.16)

Таблица 35.16. Регистр DZRPER

Номер разряда	Условное обозначение	Описание	Доступ	Исходное состояние
[31:10]	–	Резерв		
[9:0]	PER	Определяет длительность задержки для формирования запрещенной зоны после переднего фронта	R/W	0

35.2.5.3 Регистр периода задержки после заднего фронта блока Dead zone DZFPER (Таблица 35.17)

Таблица 35.17. Регистр DZFPER

Номер разряда	Условное обозначение	Описание	Доступ	Исходное состояние
[31:10]	-			
[9:0]	PER	Определяет длительность задержки для формирования запрещенной зоны после заднего фронта	R/W	0

35.2.6 Регистры блока Chopper

35.2.6.1 Регистр управления блока Chopper CHCTL (Таблица 35.18)

Таблица 35.18. Регистр CHCTL

Номер разряда	Условное обозначение	Описание	Доступ	Исходное состояние
[31:11]	–	Резерв		
[10:8]	CHDUTY	Коэффициент заполнения дробящего сигнала 0h – 1/8 1h – 2/8 2h – 3/8 3h – 4/8 4h – 5/8 5h – 6/8 6h – 7/8 7h – не используется		

Номер разряда	Условное обозначение	Описание	Доступ	Исходное состояние
[7:5]	CHCLKDIV	Выбор частоты дробящего сигнала 0h – CLK/8 1h – CLK/16 2h – CLK/24 3h – CLK/32 4h – CLK/40 5h – CLK/48 6h – CLK/56 7h – CLK/64		
[4:1]	FIRSTWTH	Ширина первого импульса 0h – 0xCLK/8 1h – 1xCLK/8 2h – 2xCLK/8 3h – 3xCLK/8 … Fh – 15xCLK/8		
[0]	CHEN	Управление работой блока Chopper 0 – выключена функция дробления 1 – включена функция дробления		

35.2.7 Регистры блока Trip unit

35.2.7.1 Регистр выбора принципа работы блока trip unit TUSEL (Таблица 35.19)

Таблица 35.19. Регистр TUSEL

Номер разряда	Условное обозначение	Описание	Доступ	Исходное состояние
[31:16]	–	Резерв		
[15:8]	ONE7 … ONE0	Выбор используемых TU[7:0] сигналов для канала PWM работающих в режиме однократного срабатывания 0 – вход не используется 1 – вход используется в режиме однократного срабатывания	R/W	0
[7:0]	MULT7 … MULT0	Выбор используемых TU[7:0] сигналов для канала PWM работающих в режиме многократного срабатывания 0 – вход не используется 1 – вход используется в режиме многократного срабатывания	R/W	0

35.2.7.2 Регистр определения реакции на событие блока trip unit TUCTL (Таблица 35.20)

Таблица 35.20. Регистр TUCTL

Номер разряда	Условное обозначение	Описание	Доступ	Исходное состояние
[31:4]	–	Резерв		
[3:2]	TUB	Когда происходит TRIP событие, к выходу OUTB могут быть применены следующие действия: 0h – OUTB переводится в high impedance state 1h – OUTB подтягивается к 1 2h – OUTB подтягивается к 0 3h – действий не производится	R/W	0

[1:0]	TUA	Когда происходит TRIP событие, к выходу OUTA могут быть применены следующие действия: 0h – OUTA переводится в high impedance state 1h – OUTA подтягивается к 1 2h – OUTA подтягивается к 0 3h – действий не производится	R/W	0

35.2.7.3 Регистр маски прерывания блока trip unit TUINTM (Таблица 35.21)

Таблица 35.21. Регистр TUINTM

Номер разряда	Условное обозначение	Описание	Доступ	Исходное состояние
[31:2]	–	Резерв		
[1]	ONE	Разрешение прерывания при возникновении события ONE TRIP 0 – прерывание запрещено 1 – прерывание разрешено, генерируется прерывание PWMTU_INT	R/W	0
[0]	MULT	Разрешение прерывания при возникновении события MULT TRIP 0 – прерывание запрещено 1 – прерывание разрешено, генерируется прерывание PWMTU_INT	R/W	0

35.2.7.4 Регистр статуса блока trip unit TUSTS (Таблица 35.22)

Таблица 35.22. Регистр TUSTS

Номер разряда	Условное обозначение	Описание	Доступ	Исходное состояние
[31:3]	–	Резерв		
[2]	TUINT	Признак возникновения прерывания 0 – прерывания не было 1 – было сформировано прерывание PWMTU_INT Дальнейшие прерывания генерироваться не будут до тех пор, пока данный флаг не будет очищен. Если флаг прерывания будет очищен, а события его вызвавшие все еще установлены, то сформируется еще одно прерывание. Чтобы избежать дальнейшей генерации прерывания нужно очистить все флаги статуса.	R	0
[1]	ONE	Статус события ONE 0 – событие не произошло 1 – событие произошло Сброс данного бита осуществляется только программно записью «1» в регистр TUCLR[ONE]	R	0
[0]	MULT	Статус события MULT 0 – событие не произошло 1 – событие произошло Данный бит остается установленным до тех пор, пока не будет очищен. Очищается он программно (записью «1» в регистр TUCLR[MULT]) или при достижении основного счетчика нулевого значения (CTRCNT=0), но только в том случае, если в момент очистки бита не будет присутствовать событие MULT на входах TU	R	0

Номер разряда	Условное обозначение	Описание	Доступ	Исходное состояние
		(&TU == 1). Если событие вызвавшее возникновение статуса все еще присутствует, то сбрасываемый бит вновь возводится в «1»		

35.2.7.5 Регистр сброса статуса блока trip unit TUCLR (Таблица 35.23)

Таблица 35.23. Регистр TUCLR

Номер разряда	Условное обозначение	Описание	Доступ	Исходное состояние
[31:2]	–	Резерв		
[2]	TUINT	Бит сброса статуса глобального прерывания PWMTU_INT 0 – нет эффекта, читается всегда как 0 1 – очищает флаг прерывания TUSTS[TUINT] Никакие дальнейшие прерывания PWMTU_INT не будут формироваться, пока флаг не очищен. Если TUSTS[INT], бит очищен, а какой-либо из других флагов будет установлен, то будет сформирован еще один импульс прерывания. Очистка всех флагов предотвратит дальнейшее формирование запросов на прерывание.	R/W	0
[1]	ONE	Бит сброса статуса события ONE 0 – нет эффекта, читается всегда как 0 1 – очищает этот флаг	R/W	0
[0]	MULT	Бит сброса статуса события MULT 0 – нет эффекта, читается всегда как 0 1 – очищает этот флаг	R/W	0

35.2.7.6 Регистр программного формирования событий блока Trip unit TUFRC (Таблица 35.24)

Таблица 35.24. Регистр TUFRC

Номер разряда	Условное обозначение	Описание	Доступ	Исходное состояние
[31:2]	–	Резерв		
[1]	ONE	Программная эмуляция возникновения события ONE TRIP 0 – запись 0 игнорируется, читается всегда 0 1 – запись 1 имитирует возникновение события ONE TRIP и провоцирует установку бита TUSTS[ONE]	R/W	0
[0]	MULT	Программная эмуляция возникновения события MULT TRIP 0 – запись 0 игнорируется, читается всегда 0 1 – запись 1 имитирует возникновение события MULT TRIP и провоцирует установку бита TUSTS[MULT]	R/W	0

35.2.8 Регистры блока Interrupt control

35.2.8.1 Регистр выбора принципа работы блока Interrupt control ICSEL (Таблица 35.25)

Таблица 35.25. Регистр ICSEL

Номер разряда	Условное обозначение	Описание	Доступ	Исходное состояние
[31:4]	–	Резерв		
[3]	INTEN	Бит разрешения формирования прерывания PWM_INT 0 – прерывание PWM_INT запрещено 1 – прерывание PWM_INT разрешено	R/W	0
[2:0]	INTSEL	Выбор источника прерывания 0h – 1h – резерв 2h – признак равенства счетчика нулю CTRCNT=0 3h – признак равенства счетчика периоду счета CTRCNT=CTRPRD 4h – признак равенства счетчика регистру сравнения CMPA во время инкремента 5h – признак равенства счетчика регистру сравнения CMPA во время декремента 6h – признак равенства счетчика регистру сравнения CMPB во время инкремента 7h – признак равенства счетчика регистру сравнения CMPB во время декремента	R/W	0

35.2.8.2 Регистр выбора принципа работы блока Interrupt control ICCTL (Таблица 35.26)

Таблица 35.26. Регистр ICCTL

Номер разряда	Условное обозначение	Описание	Доступ	Исходное состояние
[31:4]	–	Резерв		
[3:2]	EVENTCNT	Счетчик событий вызывающих прерывание. Эти биты отображают количество произошедших событий выбранных регистром ICSEL[INTSEL]. Данный счетчик автоматически сбрасывается при формировании импульса прерывания. Если прерывание отключено ICSEL[INTEN]=0 или уже сформировано ICSTS[INT]=1, то подсчет событий не производится 0h – события не обнаружены 1h – обнаружено 1 событие 2h – обнаружено 2 события 3h – обнаружено 3 события	R	0
[1:0]	EVENTPRD	Выбор периода прерываний PWM_INT. Эти биты определяют, сколько необходимо зафиксировать событий, выбранных регистром ICSEL[INTSEL] для формирования прерывания. 0h – нет генерации прерывания 1h – генерация прерывания каждый раз при возникновении события 2h – генерация прерывания каждый второй раз при возникновении события	R/W	0

Номер разряда	Условное обозначение	Описание	Доступ	Исходное состояние
		3h - генерация прерывания каждый третий раз при возникновении события		

35.2.8.3 Регистр статуса блока Interrupt control ICSTS (Таблица 35.27)

Таблица 35.27. Регистр ICSTS

Номер разряда	Условное обозначение	Описание	Доступ	Исходное состояние
[31:1]	–	Резерв		
[0]	INT	Отображает состояние входа PWM_INT 0 – нет прерывания 1 – есть прерывание. Новые прерывания не формируются до тех пор, пока данный бит не будет очищен.	R	0

35.2.8.4 Регистр очистки битов статуса блока Interrupt control ICCLR (Таблица 35.28)

Таблица 35.28. Регистр ICCLR

Номер разряда	Условное обозначение	Описание	Доступ	Исходное состояние
[31:1]	–	Резерв		
[0]	INT	Бит сброса прерывания PWM_INT 0 – нет действия, читается всегда 0 1 – сброс прерывания PWM_INT и его статусного бита	R/W	0

35.2.8.5 Регистр программного формирования событий блока Interrupt control ICFRC (Таблица 35.29)

Таблица 35.29. Регистр ICFRC

Номер разряда	Условное обозначение	Описание	Доступ	Исходное состояние
[31:1]	–	Резерв		
[0]	INT	Бит программного формирования прерывания 0 – нет действия, читается всегда 0 1 – запись 1 ведет к формированию прерывания PWM_INT и установке его статусного бита	R/W	0

35.2.9 Регистр REG

Регистр REG.

Формат регистра REG приведен в Таблица 35.30.

Таблица 35.30. Формат регистра REG

Номер бита	Условное обозначение	Назначение
0	FIELD	Бит
31:0	–	Резерв

35.3 Функционирование

Структурная схема PWM представлена на Рисунок 35.1

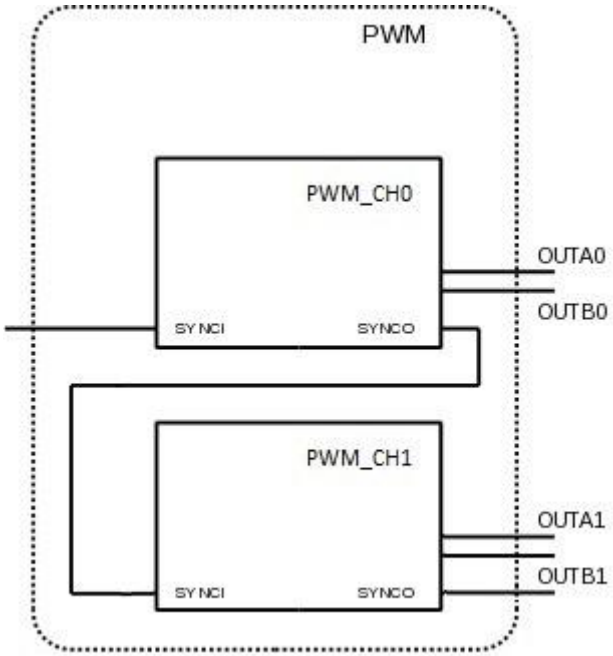


Рисунок 35.1. Структурная схема PWM

PMW_CH состоит из 8 блоков (Рисунок 35.2)

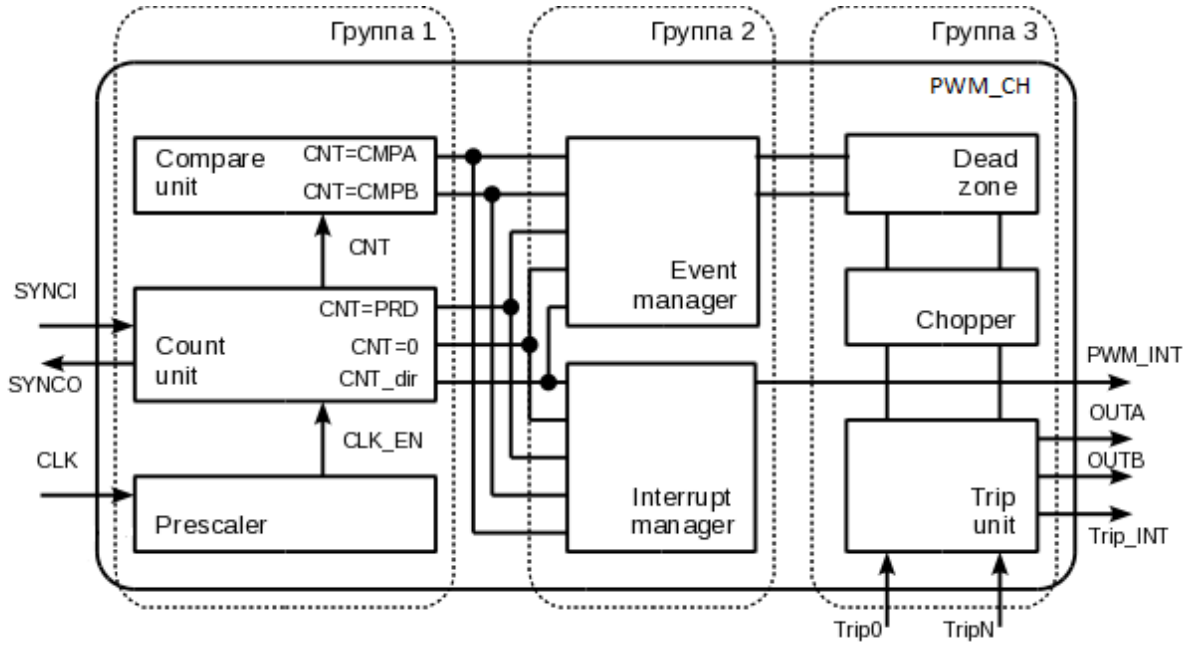


Рисунок 35.2. Структурная схема PWM_CH

Условно их можно разделить на три группы:

Группа 1 – основные функциональные блоки (предделитель, счетчик, компаратор). Основная функция данной группы – формирование ряда периодичных событий, на основе которых осуществляется функционирование всего канала.

Группа 2 – блоки определяющие реакцию на события, поступающие из первой группы блоков и формирующие основные сигналы PWM_CH OUTA, OUTB, PWM_INT. Для широтно-импульсной модуляции достаточно задействовать первые две группы блоков.

Группа 3 – блоки, осуществляющие дополнительное форматирование сформированных на предыдущих стадиях сигналов. Данная группа блоков может использоваться по отдельности, все вместе, так и не использоваться вообще, в этом случае сигналы проходят на выход без изменения.

Для большей функциональной гибкости используется структура из двух независимых PWM_CH устройств с возможностью одновременного пуска.

35.3.1 Предделитель частоты (prescaler)

На Рисунок 35.3 представлена функциональная схема предделителя частоты.

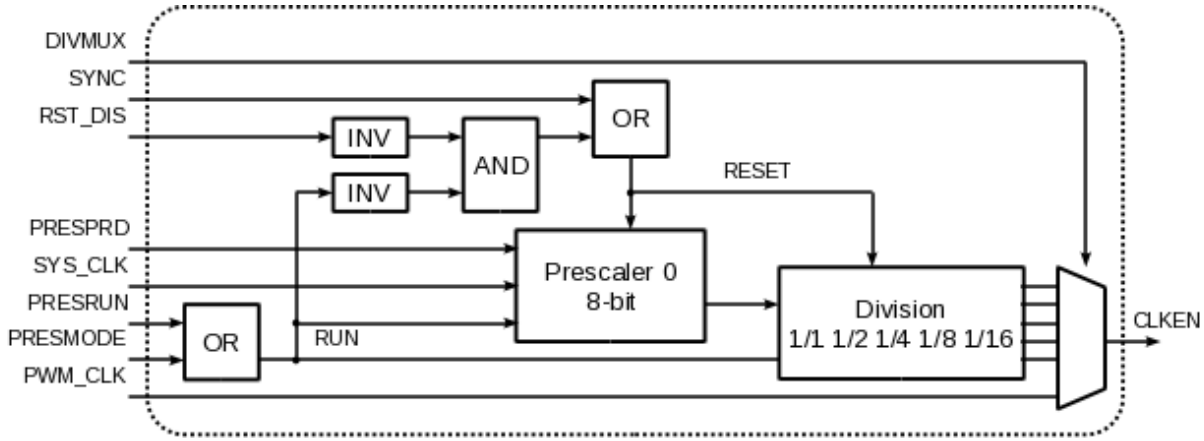


Рисунок 35.3 Функциональная схема предделителя частоты

Задача предделителя – подготовить тактовый сигнал нужного периода (CLKEN). Генерация данного сигнала может осуществляться постоянно (в режиме PRESMODE == 1) или только при включении PWM_CH (PRESMODE == 0 & PRESRUN = 1, где PRESRUN – признак включения PWM_CH).

Во время остановки таймера предделитель переходит в исходное состояние (при RST_DIS == 0) или сохраняет текущее (при RST_DIS == 1)

Данная система пуска и останова позволяет при наращивании количества PWM_CH в системе реализовать их синхронную работу.

Перевод предделителя в исходное состояние возможен также при возникновении внешнего сигнала SYNCI (или его программной эмуляции).

Для формирования рабочей частоты используются два уровня деления. На первом уровне (prescaler) реализован 8-разрядный счетчик-предделитель способный уменьшить частоту в 256 раз (в соответствии со значением регистра CLKCTL[PRESPRD]), на втором уровне частота делится в соотношении 1/1, 1/2, 1/4, 1/8, 1/16 либо переключается на внешний источник тактового сигнала PWM_CLK (в соответствии со значением регистра CLKCTL[DIVMUX]).

Для управления пуском предделителей различных каналов PWM Рисунок 35.4 используется единый регистр CTRRUN позволяющий синхронизировать работу таймеров в момент пуска и останова.

В PWM реализовано 2-х независимых канала, каждый из которых может работать как асинхронно друг относительно друга, так и синхронно.

Настройка периода частоты синхронизации таймера осуществляется программированием регистра CLKCTL полей PRESPRD и DIVMUX.

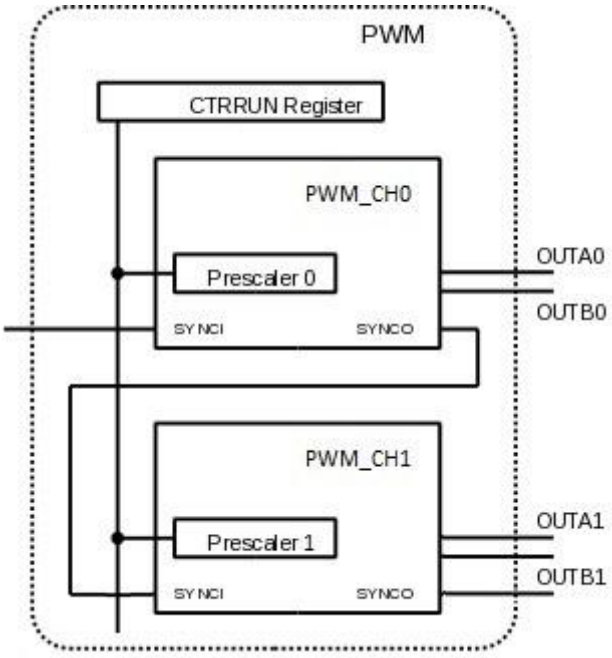


Рисунок 35.4. Управление пуском предделителей различных каналов PWM

Период полученного сигнала CLKENx можно определить по формуле:

$T_{CLKEN} = T_{CLK} * (CLKCTL[PRESPRD] + 1) * 2^{CLKCTL[DIVMUX]}$ - при CLKCTL[DIVMUX] ≤ 4

$T_{CLKEN} = 2 * T_{PWM\_CLK}$ - при CLKCTL[DIVMUX] > 4

35.3.2 Основной счетчик (Count unit)

Функции:

- организует счет в трех режимах:
  - up;
  - down;
  - up-down;
  - генерирует события:
  - равенство счетчика нулю (CTRCNT=0);
  - равенство счетчика периоду (CTRCNT=CTRPRD);
- синхронизация фазы с другими счетчиками.

Вычисление периода и частоты PWM_CH (Рисунок 35.5)

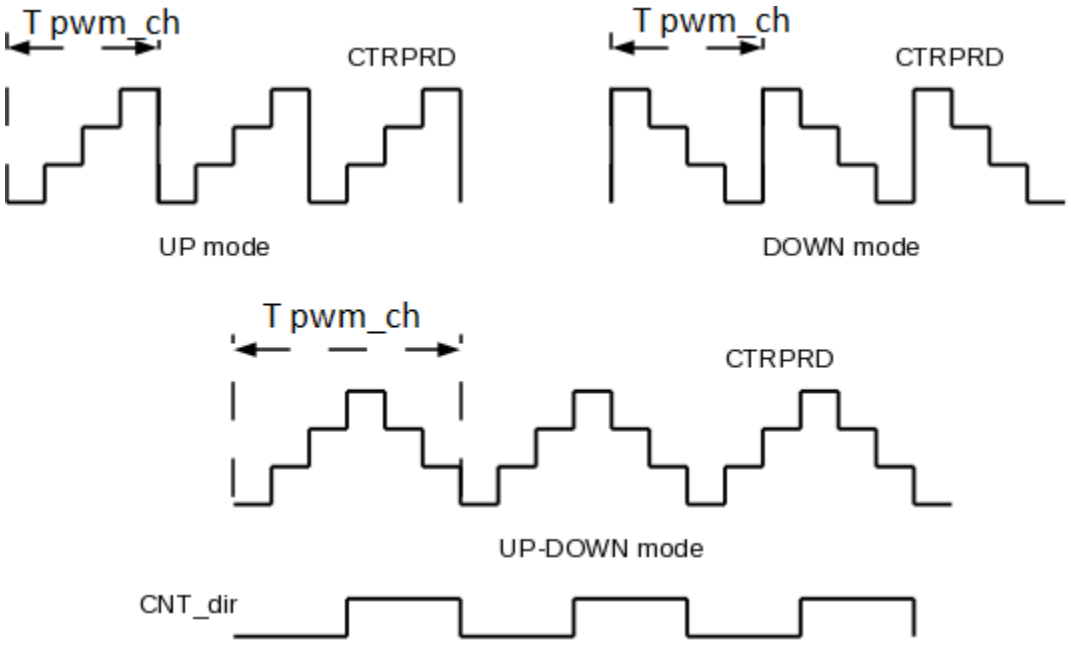


Рисунок 35.5. Вычисление периода и частоты PWM_CH

Таким образом период PWM_CH определяем по формуле $T_{PWM\_CH} = (CTRPRD + 1) * T_{CLKEN}$ - в режимах UP и DOWN, $T_{PWM\_CH} = 2 * (CTRPRD + 1) * T_{CLKEN}$ - режиме UP-DOWN.

Теневой (буферный) регистр периода CTRPRD

Регистр периода CTRPRD имеет двойника, который используется для аппаратной синхронизации записи нового значения периода. Фактически имеется два регистра:

- активный регистр – регистр непосредственно участвующий в работе счетчика;
- теневой регистр – регистр, предназначенный для временного хранения данных предназначенных для активного регистра.

Адрес у этих регистров один и тот же, а с каким из них работает программист в данный момент определяет бит CLKCTL[LOADPRD]:

- режим работы с участием теневого регистра CLKCTL[LOADPRD]=0. Данные записанные программистом в регистр периода хранятся в теневом регистре до тех пор, пока счетчик не будет равен нулю (CTRCNT=0). Когда счетчик обнулится данные из теневого регистра перепишутся в активный;

- режим работы без участия теневого регистра CLKCTL[LOADPRD]=1. Доступ к активному регистру осуществляется напрямую. Данный режим установлен по умолчанию.

Синхронизация каналов PMW при помощи сигналов SYNCI/O (Рисунок 35.6)

Под синхронизацией понимается перевод счетчика CTRCNT в значение, прописанное в регистре фазы синхронизации CTRPHS. Синхронизацию можно отключить при помощи бита CLKCTL[SYNCPHSEN]. Признаком к старту синхронизации может быть:

- импульс на входе SYNCI

- программная синхронизация (запись бита CLKCTL[SWFSYNC])

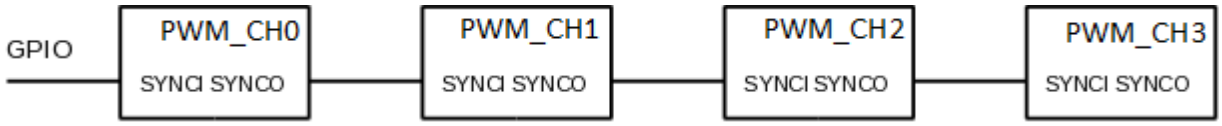


Рисунок 35.6. Синхронизация каналов

Примеры синхронизации.

Таймер работает в режиме UP-COUNT (Рисунок 35.7).

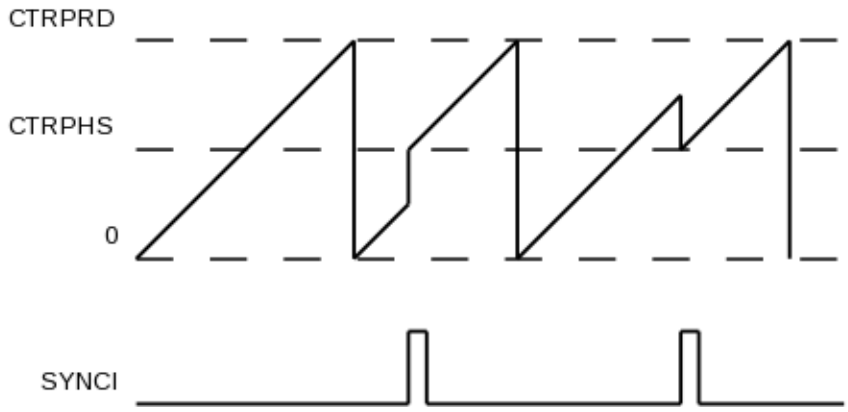


Рисунок 35.7. Синхронизация при работе в режиме UP_COUNT

Таймер работает в режиме DOWN-COUNT (Рисунок 35.8).

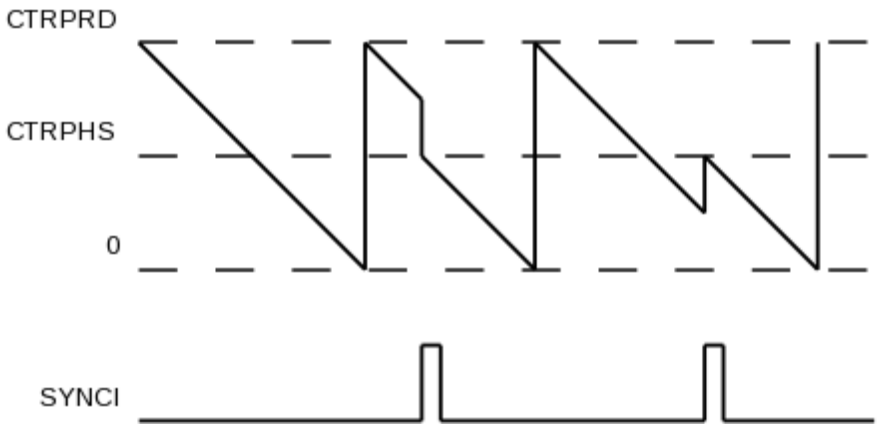


Рисунок 35.8. Синхронизация при работе в режиме DOWN_COUNT

Таймер работает в режиме UP-DOWN-COUNT (CLKCTL[DIRSYNC]=0 – после синхронизации счетчик декрементируется Рисунок 35.9).

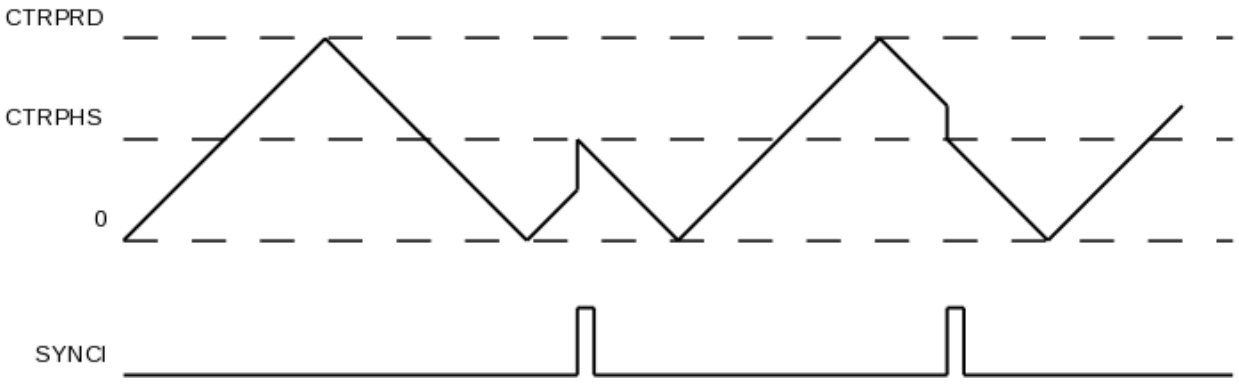


Рисунок 35.9. Синхронизация при работе в режиме UP-DOWN-COUNT

Таймер работает в режиме UP-DOWN-COUNT (CLKCTL[DIRSYNC]=0 – после синхронизации счетчик инкрементируется Рисунок 35.10)

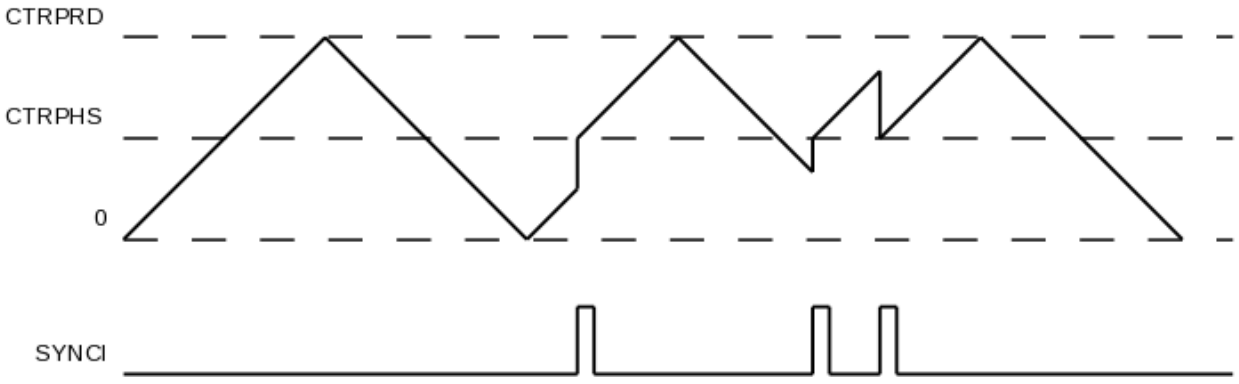


Рисунок 35.10. Синхронизация при работе в режиме UP-DOWN-COUNT

Пуск и остановка счетчиков осуществляется записью соответствующего значения в регистр CTRRUN[RUN*] (Рисунок 35.11). Для управления каналами PWM (PWM_CH0 –

PWM_CH1) в этом регистре выделены по 3 бита. Наличие основных управляющих сигналов в одном регистре позволяет синхронно управлять всеми каналами.

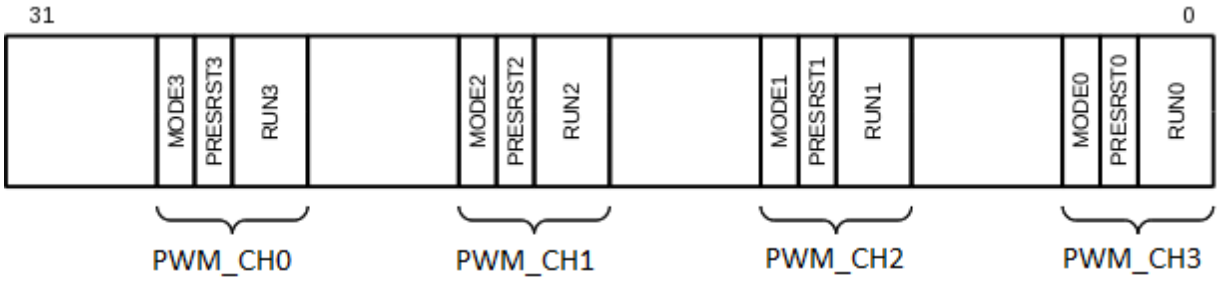


Рисунок 35.11. Распределение управляющих битов в регистре CTRRUN

При пуске таймера основной счетчик (CTRCNT) начинает счет с того состояния в котором он находится. В том случае, если счетчик запускается в режиме «DOWN-COUNT», а CTRCNT=0, то следующее значение CTRCNT=CTRPRD, при этом сформируется признак события равенства счетчика нулю ZERO_EVENT.

Для отображения направления счета используется статусный бит CTRSTS[DIR_STS].

При пуске канала в режиме «DOWN-COUNT» DIR_STS устанавливается в 0, в остальных случаях устанавливается в 1. При работе в режиме «UP-DOWN-COUNT» DIR_STS инвертируется при достижении счетчика нуля (CTRCNT=0) и периода счета (CTRCNT=CTRPRD).

Для удобства статусные биты от всех каналов PWM также сосредоточены в одном регистре CTRSTS (Рисунок 35.12).

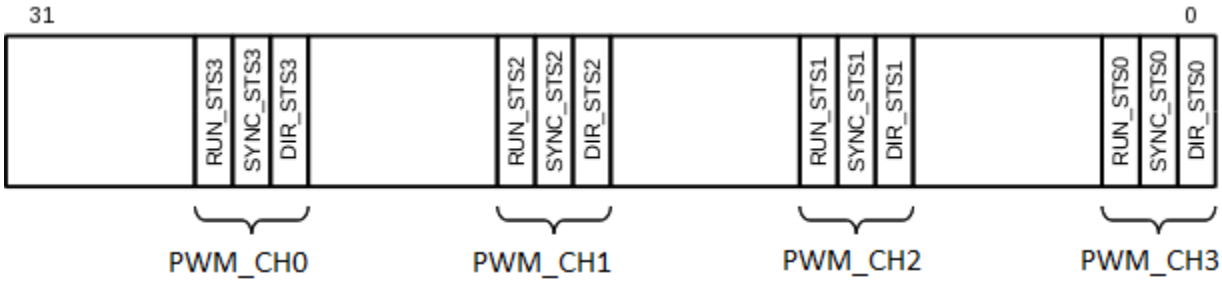


Рисунок 35.12. Распределение битов статуса в регистре CTRSTS

35.3.3 Блок сравнения Compare unit

Основная задача блока – сравнение значения, записанного в регистрах CMPA и CMPB, со значением счетчика CTRCNT и формирование признаков равенства:

- CRTCNT=CMPA;

- CTRCNT=CNTB.

Регистры CMPA и CMPB имеют двойников (теневые регистры), и все операции чтения/записи по умолчанию осуществляются через них. Момент загрузки данных из теневых регистров в основные определяется в регистре управления CMPCTL[LDBMODE] для регистра CMPB и CMPCTL[LDAMODE] для регистра CMPA. Также в регистре управления присутствуют флаги, определяющие были загружены данные из теневых регистров или нет (CMPCTL[SCMPBFULL] и CMPCTL[SCMPAFULL]). При необходимости теневые регистры можно отключить и работать напрямую с CMPA и CMPB.

Примеры формирования событий совпадения при работе счетчика в различных режимах.

Таймер работает в режиме UP-COUNT (Рисунок 35.13)

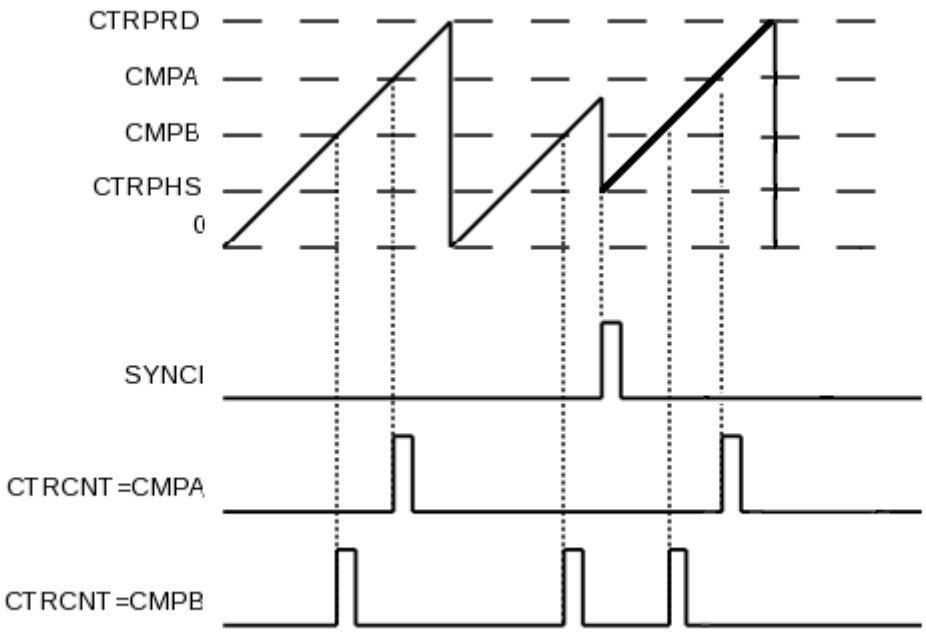


Рисунок 35.13. Формирование событий при работе в режиме UP-COUNT

Таймер работает в режиме DOWN-COUNT (Рисунок 35.14).

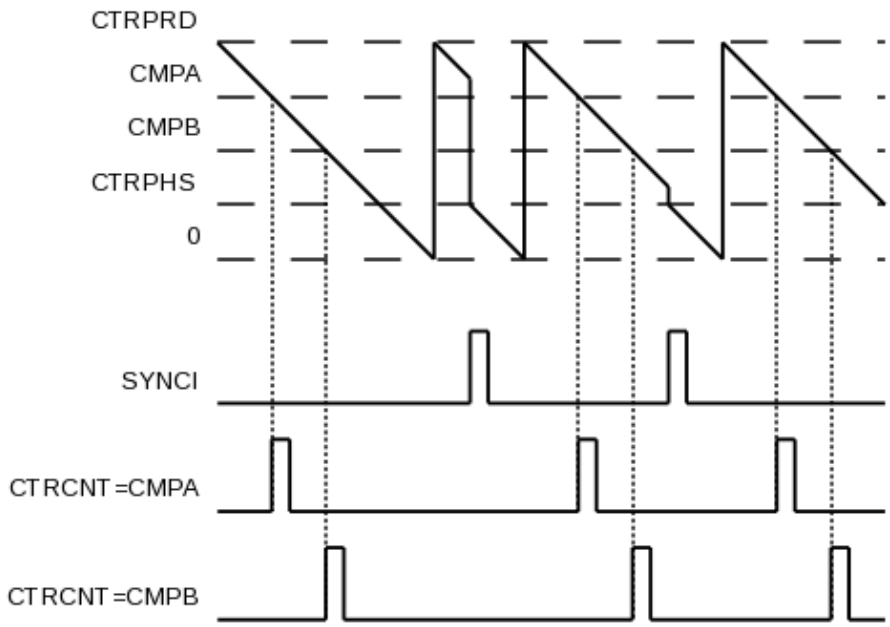


Рисунок 35.14. Формирование событий при работе в режиме DOWN-COUNT

Таймер работает в режиме UP-DOWN-COUNT (CLKCTL[DIRSYNC]=0 – после синхронизации счетчик декрементируется (Рисунок 35.15).

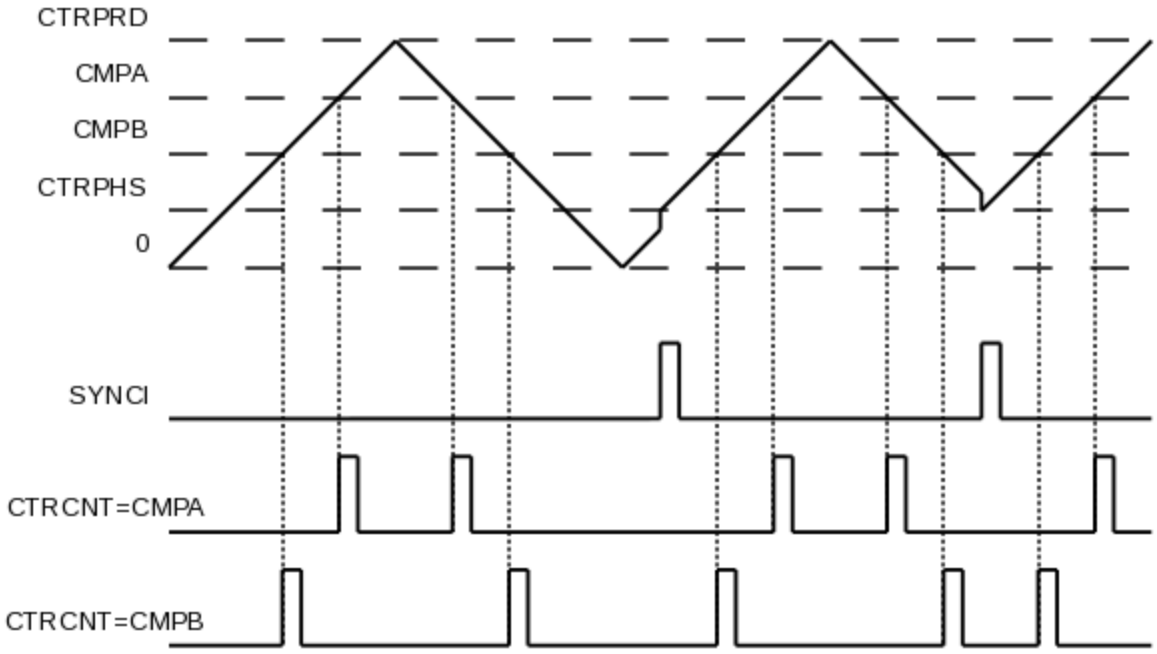


Рисунок 35.15. Формирование событий при работе в режиме UP-DOWN-COUNT

Таймер работает в режиме UP-DOWN-COUNT (CLKCTL[DIRSYNC]=0 – после синхронизации счетчик инкрементируется Рисунок 35.16).

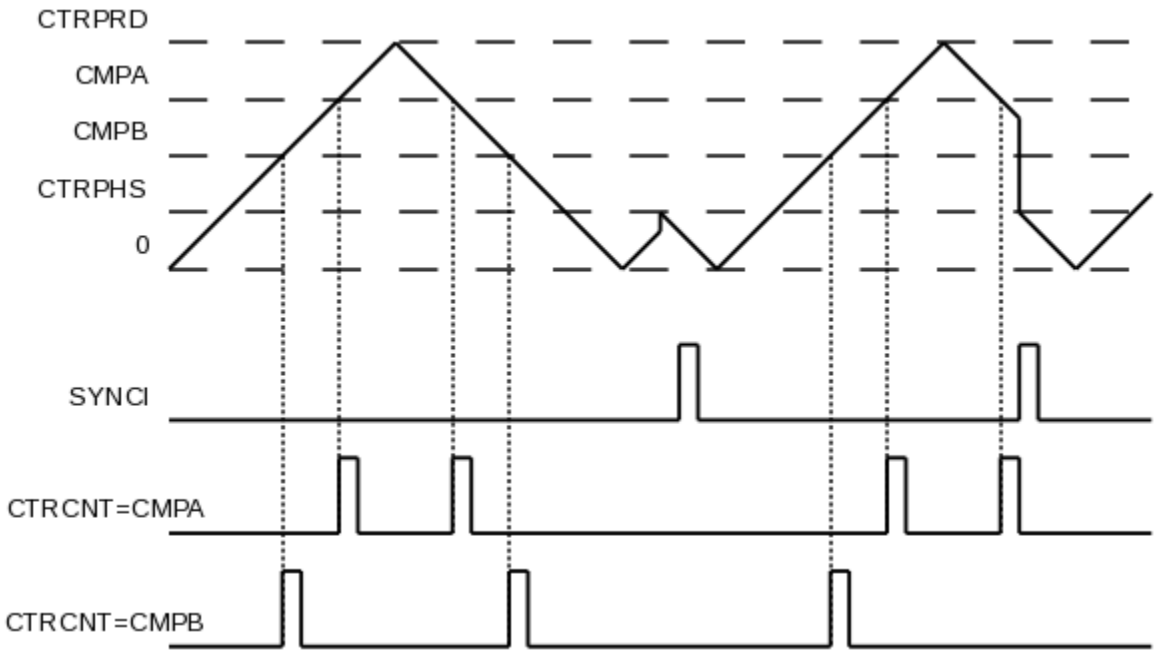


Рисунок 35.16. Формирование событий при работе в режиме UP-DOWN-COUNT

35.3.4 Блок реакции на событие Event manager

Задача данного блока – формирование выходных сигналов OUTA и OUTB на основе следующих событий:

- CTRCNT = PRD;
- CTRCNT = CMPA;
- CTRCNT = CMPB;
- CTRCNT = 0.
- программное переключение (SW forced)

При возникновении любого из этих событий возможна одна из следующих реакций:

- установить OUTA и/или OUTB в 1;
- установить OUTA и/или OUTB в 0;
- инвертировать OUTA и/или OUTB.
- OUTA и/или OUTB оставить без изменений.

Как реагировать на возникающие события определяется регистрами EMCTLA для OUTA и EMCTLB для OUTB.

Регистры EMSWFR и EMCSWFR определяют реакцию на программное переключение.

При одновременном возникновении событий отрабатывается событие с наивысшим приоритетом.

Для каждого из режимов работы счетчика определен порядок обработки событий.

Таймер работает в режиме UP-COUNT (перечислено в порядке убывания приоритета Таблица 35.31)

Таблица 35.31. Порядок обработки событий при работе в UP-COUNT режиме

1	SW forced
2	CTRCNT = PRD
3	CTRCNT = CMPB
4	CTRCNT = CMPA
5	CTRCNT = 0

Таймер работает в режиме DOWN-COUNT (Таблица 35.32).

Таблица 35.32. Порядок обработки событий при работе в DOWN-COUNT режиме

1	SW forced
2	CTRCNT = 0
3	CTRCNT = CMPB
4	CTRCNT = CMPA
5	CTRCNT = PRD

Таймер работает в режиме UP-DOWN-COUNT (Таблица 35.33).

Таблица 35.33. Порядок обработки событий при работе в UP-DOWN-COUNT режиме

	Счетчик инкрементируется	Счетчик декрементируется
1	SW forced	SW forced
2	CTRCNT = CMPB	CTRCNT = CMPB
3	CTRCNT = CMPA	CTRCNT = CMPA
4	CTRCNT = 0	CTRCNT = PRD

Пример формирования выходных сигналов OUTA и OUTB при работе счетчика в режиме UP-DOWN (Рисунок 35.17).

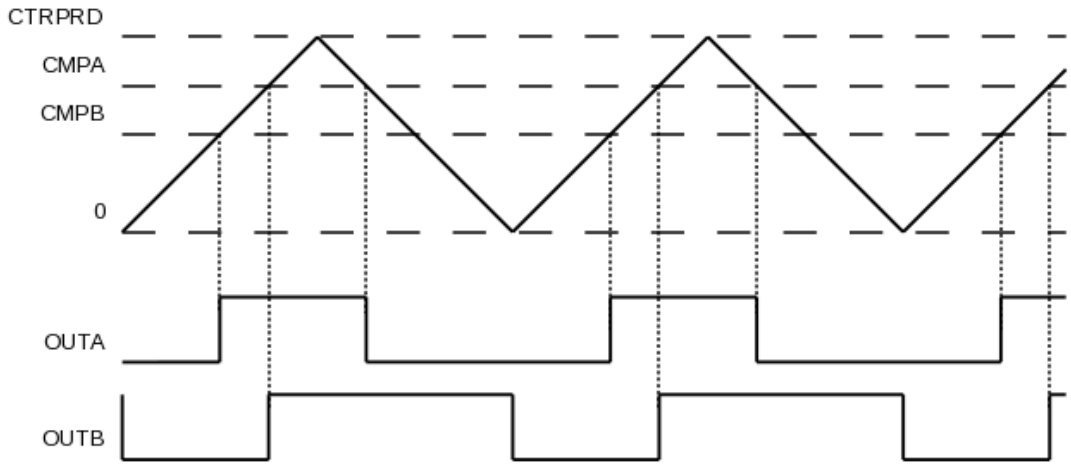


Рисунок 35.17. Пример формирования выходных сигналов OUTA и OUTB

35.3.5 Генератор запретной зоны (Dead zone)

Блок реакции на события позволяет достаточно гибко управлять формой выходного сигнала, но для формирования определенной задержки после фронта сигнала более удобно использовать генератор запретной зоны.

Функциональная схема блока представление на Рисунок 35.18**.**

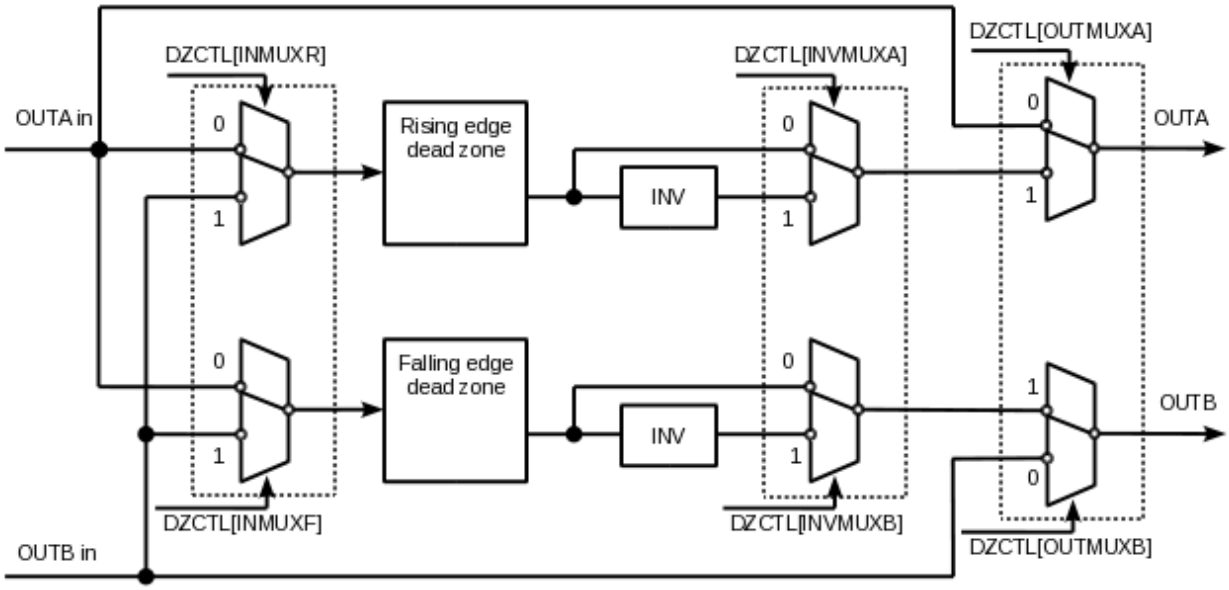


Рисунок 35.18. Функциональная схема блока генерации запретной зоны

Основная функция блока – формирование «мертвой» зоны (формирование выходного сигнала со сдвигом фронта относительно входного сигнала) после переднего, заднего или обоих фронтов.

Для управления режимом работы блока используется регистр DZCTL. Задержки после переднего и заднего фронтов задаются в регистрах DZRPER и DXFPER соответственно.

Пример генерации запретной зоны для сигнала “OUTA in” от переднего и заднего фронтов без выходной инверсии (Рисунок 35.19).

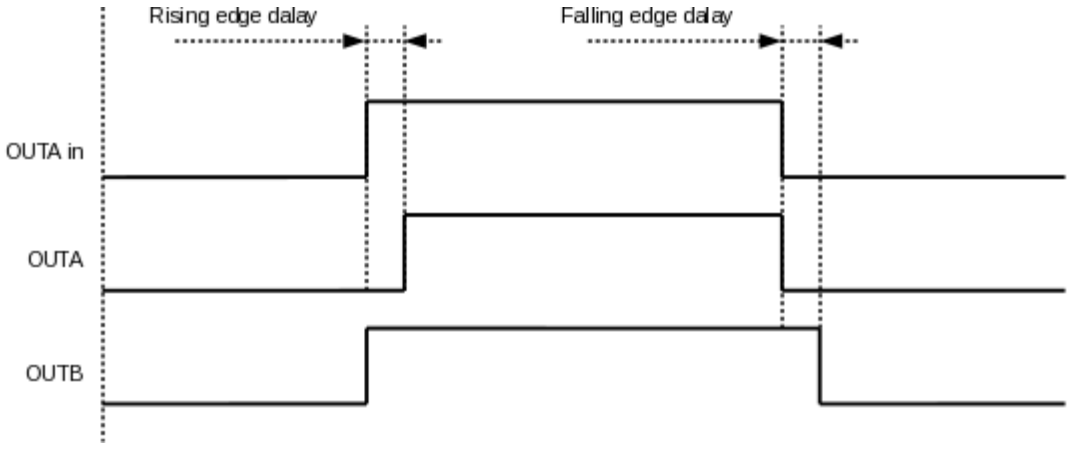


Рисунок 35.19. Пример генерации запретной зоны

Величину задержки после фронта можно определить по формуле:

T_{CLKEN} * DZRPER – для переднего фронта (rising edge delay);

T_{CLKEN} * DZFPER – для заднего фронта (falling edge delay)

35.3.6 Блок дробления выходного сигнала Chopper

Основная функция блока – наложение на выходной сигнал дробящей частоты.

Поддерживаются следующие возможности:

- управление длительностью первого импульса (CHCTL[FIRSTWTH]);
- управление периодом дробящей частоты (CHCTL[CHCLKDIV]);
- управление коэффициентом заполнения дробящей частоты (CHCTL[CHDUTY]);
- выключение данной функции.

Пример простого наложения дробящей частоты (Рисунок 35.20).

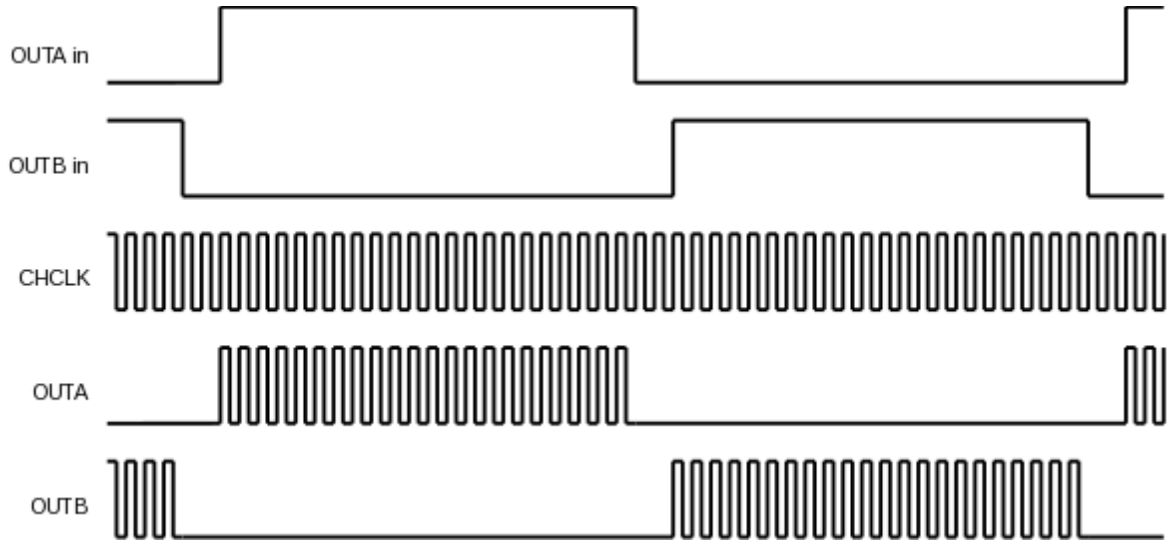


Рисунок 35.20. Пример наложения дробящей частоты

Пример наложения дробящей частоты и первого импульса (Рисунок 35.21).

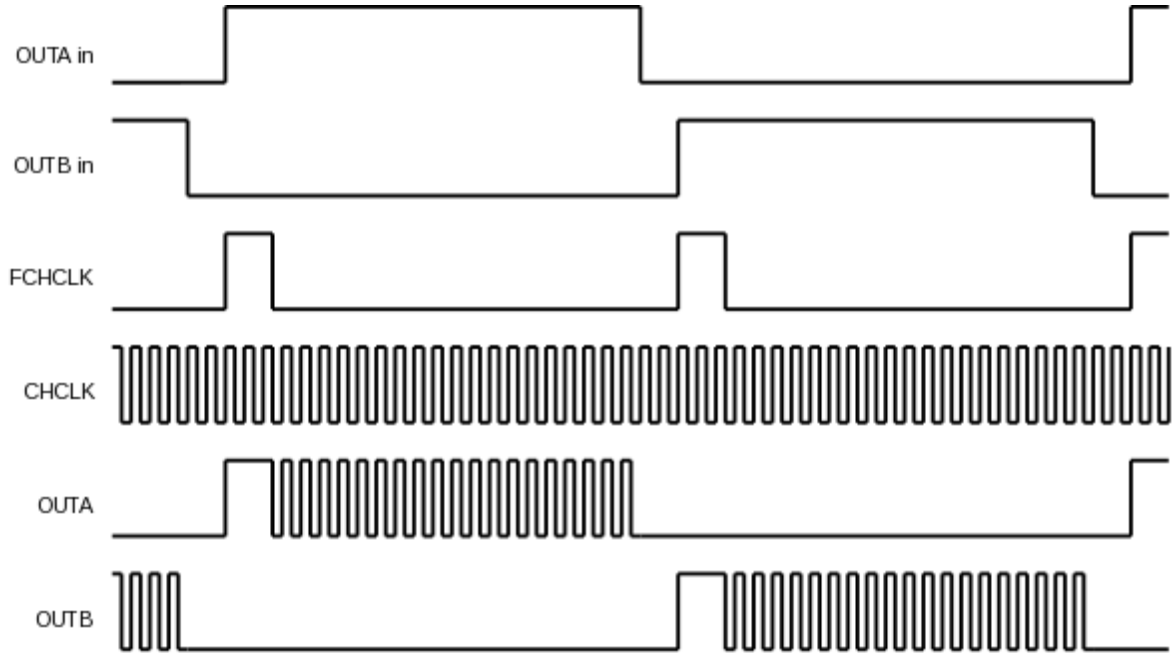


Рисунок 35.21. Пример наложения дробящей частоты и первого импульса

35.3.7 Блок реакции на внешнее воздействие Trip unit

В следующем разделе описывается возможные операции и конфигурационные параметры блока Trip unit.

Основная функция блока – анализ входных воздействий по линиям TU и реакция на них путем изменения сигналов OUTA и OUTB, а также формирование запроса на прерывание процессора по каналу PWMTU_INT.

Входные сигналы TU0 – TU7 активны низким уровнем. При установке на одном из данных входов «0» блок выполняет определенное действие. Каждый канал PWM может быть индивидуально настроен на каждый из TU* сигналов. Какие линии TU будут использоваться конкретным каналом PWM, программируется в регистре TUSEL этого блока. TU сигнал может быть как синхронным, так и асинхронным относительно CLK.

TU* вход может быть индивидуально сконфигурирован, чтобы обеспечить как цикличную реакцию на данное событие, так и единичную. Конфигурация определяется битами TUSEL[ONE] и TUSEL[MULT].

Независимо от того в какой конфигурации мы работаем реакция на событие производится в соответствии с данными в регистре TUCTL (Таблица 35.34).

Таблица 35.34. Возможная реакция на внешнее воздействие

TUCTL[TUA] и/или TUCTL[TUB]	OUTA и/или OUTB
0h	High-impedance
1h	Подтянуть к 1
2h	Подтянуть к 0

3h	Нет действия

В режиме реакции на каждое событий (MULT) устанавливается флаг данного события TUSTS[MULT], а также генерируется прерывание PWMTU_INT если оно разрешено в регистре TUINTM[MULT].

Данное состояние автоматически сбрасывается при достижении счетчика нулевого значения CTRCNT=0 если на TU* к этому времени восстановилось единичное значение. Таким образом, в данном режиме событие сброса очищается каждый цикл счетчика PWM. TUSTS[MULT] также может быть сброшен записью 1 в TUCLR[MULT], при условии, что на TU* к этому времени восстановилось единичное значение. Если на входе TU присутствует 0, то очистить бит TUSTS[MULT] невозможно.

В режиме однократной реакции на событие (ONE) устанавливается флаг данного события TUSTS[ONE], а также генерируется прерывание PWMTU_INT если оно разрешено в регистре TUINTM[ONE]. Реакция на данное событие может быть сброшена только вручную записью 1 в TUCLR[ONE].

Запись единичных значений в поля ONE и MULT регистра TUFRC позволяет имитировать возникновение событий ONE и MULT соответственно.

35.3.8 Блок формирования запроса прерывание процессора Interrupt control

Формирует сигнал прерывания PWN_INT при возникновении различных событий. При помощи регистра TUSEL можно выбрать, на какое из следующих событий реагировать:

- CTRCNT = 0;
- CTRCNT = CTRPRD;
- CTRCNT = CMPA при инкременте счетчика;
- CTRCNT = CMPA при декременте счетчика;
- CTRCNT = CMPB при инкременте счетчика;
- CTRCNT = CMPB при декременте счетчика.

В зависимости от периода прерывания записанного в регистре ICCTL[EVENTPRD] происходит реакция:

- на каждое событие;
- на каждое второе событие;
- на каждое третье событие.

При необходимости возможно программное формирование прерывания при помощи регистра ICFRC.

36. КОНТРОЛЛЕР УНИВЕРСАЛЬНЫХ ВЫВОДОВ (GPIO)

36.1 Общая информация

Контроллер GPIO имеет следующие характеристики:

1. 32 независимо управляемых сигнала, 4 независимо управляемых порта
2. Независимое управление состоянием и направлением (в том числе и hardware режим) для каждого сигнала
3. Поддержка прерываний для порта A – 8 независимых сигналов (для каждого контроллера gpio0 и gpio1)
4. Встроенная логика фильтрации шумов

36.2 Регистры блока управления GPIO

При работе с блоком GPIO1 надо помнить о необходимости предварительной конфигурации соответствующих КП через системные регистры lsperiph1_subs.

В Таблица 36.1 приведен перечень программно-доступных регистров блока управления GPIO.

Таблица 36.1. Перечень программно-доступных регистров блока управления GPIO

Условное обозначение регистра	Название регистра	Тип доступа	Исходное Состояние	Смещение
gpio_swporta_dr	Регистр данных порта A.	W/R	0x0	0x00
gpio_swporta_ddr	Регистр направления потока данных через порт A.	W/R	0x0	0x04
gpio_swporta_ctl	Регистр источника данных порта A.	W/R	0x0	0x08
gpio_swportb_dr	Регистр данных порта B.	W/R	0x0	0x0C
gpio_swportb_ddr	Регистр направления потока данных через порт B.	W/R	0x0	0x10
gpio_swportb_ctl	Регистр источника данных порта B.	W/R	0xFFFF	0x14
gpio_swportc_dr	Регистр данных порта C.	W/R	0x0	0x18
gpio_swportc_ddr	Регистр направления потока данных через порт C.	W/R	0x0	0x1C
gpio_swportc_ctl	Регистр источника данных порта C.	W/R	0x0	0x20
gpio_swportd_dr	Регистр данных порта D.	W/R	0x0	0x24
gpio_swportd_ddr	Регистр направления потока данных через порт D.	W/R	0x0	0x28
gpio_swportd_ctl	Регистр источника данных порта D.	W/R	0x0	0x2C
gpio_inten	Регистр включения прерываний.	W/R	0x0	0x30
gpio_intmask	Регистр маскирования прерываний.	W/R	0x0	0x34

Условное обозначение регистра	Название регистра	Тип доступа	Исходное Состояние	Смещение
gpio_inttype_level	Регистр уровня прерываний.	W/R	0x0	0x38
gpio_int_polarity	Регистр полярности прерываний.	W/R	0x0	0x3C
gpio_intstatus	Регистр статуса прерываний	R	0x0	0x40
gpio_raw_intstatus	Регистр статуса немаскированных прерываний	R	0x0	0x44
gpio_debounce	Регистр включения фильтра дребезга.	W/R	0x0	0x48
gpio_porta_eoi	Регистр сброса прерываний.	W	0x0	0x4C
gpio_ext_porta	Внешний регистр порта A.	R	0x0	0x50
gpio_ext_portb	Внешний регистр порта B.	R	0x0	0x54
gpio_ext_portc	Внешний регистр порта C.	R	0x0	0x58
gpio_ext_portd	Внешний регистр порта D.	R	0x0	0x5C

36.2.1 Регистр gpio_swporta_dr

Регистр данных порта A.

Формат регистра gpio_swporta_dr приведен в Таблица 36.2.

Таблица 36.2. Формат регистра gpio_swporta_dr

номер бита	Условное обозначение	Назначение
31:0	gpio_swporta_dr	Значение, записанное в этот регистр, передается через выходной сигнал порта A, если соответствующие биты регистра направления потока данных порта A (gpio_swporta_ddr) установлены на режим выдачи данных, а соответсвющие биты регистра источника данных порта A (gpiop_swporta_ddr) установлены на программный источник. Чтение данных из этого регистра возвращает последние записанные в этот регистр данные.

36.2.2 Регистр gpio_swporta_ddr

Регистр направления потока данных через порт A.

Формат регистра gpio_swporta_ddr приведен в Таблица 36.3.

Таблица 36.3. Формат регистра gpio_swporta_ddr

Номер бита	Условное обозначение	Назначение
31:0	gpio_swporta_ddr	Каждый бит этого регистра контролирует направление потока данных через соответствующий бит регистра данных порта A (gpio_swporta_dr). При этом: 0 - прием. 1 - выдача.

36.2.3 Регистр gpio_swporta_ctl

Регистр источника данных порта A.

Формат регистра gpio_swporta_ctl приведен в Таблица 36.4.

Таблица 36.4. Формат регистра gpio_swporta_ctl

Номер бита	Условное обозначение	Назначение
31:0	gpio_swporta_ctl	Каждый бит этого регистра выбирает источник данных для соответствующего сигнала порта A. Источник может быть программным (соответствующие биты регистра gpio_swporta_dr) или аппаратным: 0 - программный 1 - аппаратный.

36.2.4 Регистр gpio_swportb_dr

Регистр данных порта B.

Формат регистра gpio_swportb_dr приведен в Таблица 36.5.

Таблица 36.5. Формат регистра gpio_swportb_dr

номер бита	Условное обозначение	Назначение
31:0	gpio_swportb_dr	Значение, записанное в этот регистр, передается через выходной сигнал порта B, если соответствующие биты регистра направления потока данных порта B (gpio_swportb_ddr) установлены на режим выдачи данных, а соответствующие биты регистра источника данных порта B (gpiop_swportb_ddr) установлены на программный источник. Чтение данных из этого регистра возвращает последние записанные в этот регистр данные.

36.2.5 Регистр gpio_swportb_ddr

Регистр направления потока данных через порт B.

Формат регистра gpio_swportb_ddr приведен в Таблица 36.6.

Таблица 36.6. Формат регистра gpio_swportb_ddr

Номер бита	Условное обозначение	Назначение
31:0	gpio_swportb_ddr	Каждый бит этого регистра контролирует направление потока данных через соответствующий бит регистра данных порта B (gpio_swportb_dr). При этом: 0 - прием. 1 - выдача.

36.2.6 Регистр gpio_swportb_ctl

Регистр источника данных порта B.

Формат регистра gpio_swportb_ctl приведен в Таблица 36.7.

Таблица 36.7. Формат регистра gpio_swportb_ctl

Номер бита	Условное обозначение	Назначение
31:0	gpio_swportb_ctl	Каждый бит этого регистра выбирает источник данных для соответствующего сигнала порта B. Источник может быть программным (соответствующие биты регистра gpio_swportb_dr) или аппаратным: 0 - программный 1 - аппаратный.

36.2.7 Регистр gpio_swportc_dr

Регистр данных порта C.

Формат регистра gpio_swportc_dr приведен в Таблица 36.8.

Таблица 36.8. Формат регистра gpio_swportc_dr

номер бита	Условное обозначение	Назначение
31:0	gpio_swportc_dr	Значение, записанное в это регистр, передается через выходной сигнал порта C, если соответствующие биты регистра направления потока данных порта C (gpio_swportc_ddr) установлены на режим выдачи данных, а соответствующие биты регистра источника данных порта C (gpiop_swportc_ddr) установлены на программный источник. Чтение данных из этого регистра возвращает последние записанные в этот регистр данные.

36.2.8 Регистр gpio_swportc_ddr

Регистр направления потока данных через порт C.

Формат регистра gpio_swportc_ddr приведен в Таблица 36.9.

Таблица 36.9. Формат регистра gpio_swportc_ddr

Номер бита	Условное обозначение	Назначение
31:0	gpio_swportc_ddr	Каждый бит этого регистра контролирует направление потока данных через соответствующий бит регистра данных порта C (gpio_swportc_dr). При этом: 0 - прием. 1 - выдача.

36.2.9 Регистр gpio_swportc_ctl

Регистр источника данных порта C.

Формат регистра gpio_swportc_ctl приведен в Таблица 36.10.

Таблица 36.10. Формат регистра gpio_swportc_ctl

Номер бита	Условное обозначение	Назначение
31:0	gpio_swportc_ctl	Каждый бит этого регистра выбирает источник данных для соответствующего сигнала порта C. Источник может быть программным (соответствующие биты регистра gpio_swportc_dr) или аппаратным: 0 - программный 1 - аппаратный.

36.2.10 Регистр gpio_swportd_dr

Регистр данных порта D.

Формат регистра gpio_swportd_dr приведен в Таблица 36.11.

Таблица 36.11. Формат регистра gpio_swportd_dr

номер бита	Условное обозначение	Назначение
31:0	gpio_swportd_dr	Значение, записанное в это регистр, передается через выходной сигнал порта D, если соответствующие биты регистра направления потока данных порта D (gpio_swportd_ddr) установлены на режим выдачи данных, а соответствующие биты регистра источника данных порта D (gpiop_swportd_ddr) установлены на программный источник. Чтение данных из этого регистра возвращает последние записанные в этот регистр данные.

36.2.11 Регистр gpio_swportd_ddr

Регистр направления потока данных через порт D.

Формат регистра gpio_swportd_ddr приведен в Таблица 36.12.

Таблица 36.12. Формат регистра gpio_swportd_ddr

Номер бита	Условное обозначение	Назначение
31:0	gpio_swportd_ddr	Каждый бит этого регистра контролирует направление потока данных через соответствующий бит регистра данных порта D (gpio_swportd_dr). При этом: 0 - прием. 1 - выдача.

36.2.12 Регистр gpio_swportd_ctl

Регистр источника данных порта D.

Формат регистра gpio_swportd_ctl приведен в Таблица 36.13.

Таблица 36.13. Формат регистра gpio_swportd_ctl

Номер бита	Условное обозначение	Назначение
31:0	gpio_swportd_ctl	Каждый бит этого регистра выбирает источник данных для соответствующего сигнала порта D. Источник может быть программным (соответствующие биты регистра gpio_swportd_dr) или аппаратным: 0 - программный 1 - аппаратный.

36.2.13 Регистр gpio_inten

Регистр включения прерываний.

Формат регистра gpio_inten приведен в Таблица 36.14.

Таблица 36.14. Формат регистра gpio_inten

Номер бита	Условное обозначение	Назначение
31:0	gpio_inten	Запись 1 в любой бит этого регистра позволяет соответствующему биту порта A воспринимать входной сигнал как прерывание. Для этого также необходимо, чтобы этот бит порта A был настроен на прием данных (gpio_sporta_ddr[n] = 0), а источник данных был программным (gpio_swporta_ctl[n] = 0).

36.2.14 Регистр gpio_intmask

Регистр маскирования прерываний.

Формат регистра gpio_intmask приведен в Таблица 36.15.

Таблица 36.15. Формат регистра gpio_intmask

Номер бита	Условное обозначение	Назначение
31:0	gpio_intmask	Каждый бит этого регистра маскирует прерывание от соответствующего бита порта A: 0: прерывание не маскировано. 1: прерывание маскировано.

36.2.15 Регистр gpio_inttype_level

Регистр типа прерываний.

Формат регистра gpio_inttype_level приведен в Таблица 36.16.

Таблица 36.16. Формат регистра gpio_inttype_level

Номер бита	Условное обозначение	Назначение
31:0	gpio_inttype_level	Каждый бит этого регистра определяет тип обрабатываемого прерывания для соответствующего бита порта A:

		0: По уровню 1: По фронту

36.2.16 Регистр gpio_int_polarity

Регистр маскирования прерываний.

Формат регистра gpio_int_polarity приведен в Таблица 36.17.

Таблица 36.17. Формат регистра gpio_int_polarity

Номер бита	Условное обозначение	Назначение
31:0	gpio_int_polarity	Каждый бит этого регистр определяет полярность обрабатываемого прерывания для соответствующего бита порта A. В зависимости от типа обрабатываемого прерывания: 0: активный ноль или задний фронт. 1: активная единица или передний фронт.

36.2.17 Регистр gpio_intstatus

Регистр статуса прерываний.

Формат регистра gpio_intstatus приведен в Таблица 36.18.

Таблица 36.18. Формат регистра gpio_intstatus

Номер бита	Условное обозначение	Назначение
31:0	gpio_intstatus	Каждый бит этого регистра отображает статус прерывания для соответствующего бита порта A:

36.2.18 Регистр gpio_raw_intstatus

Регистр статуса немаскированных прерываний.

Формат регистра gpio_raw_intstatus приведен в Таблица 36.19.

Таблица 36.19. Формат регистра gpio_raw_intstatus

Номер бита	Условное обозначение	Назначение
31:0	gpio_raw_intstatus	Каждый бит этого регистра отображает статус немаскированного прерывания для соответствующего бита порта A.

36.2.19 Регистр gpio_debounce

Регистр фильтрации дребезга.

Формат регистра gpio_debounce приведен в Таблица 36.20.

Таблица 36.20. Формат регистра gpio_debounce

Номер бита	Условное обозначение	Назначение
31:0	gpio_debounce	Каждый бит этого регистра управляет фильтрующей логикой устранения дребезга входного сигнала для соответствующего бита порта A. При установке бита в 1, gpio сформирует прерывание, в случае если активный уровень входного сигнала длится 2 такта фильтрующей частоты gpio dbclk.

36.2.20 Регистр gpio_porta_eoi

Регистр сброса прерываний.

Формат регистра gpio_porta_eoi приведен в Таблица 36.21.

Таблица 36.21. Формат регистра gpio_porta_eoi

Номер бита	Условное обозначение	Назначение
31:0	gpio_porta_eoi	Запись 1 в любой бит этого регистра сбросит прерывание от соответствующего бита порта A.

36.2.21 Регистр gpio_ext_porta

Внешний регистр порта A.

Формат регистра gpio_ext_porta приведен в Таблица 36.22.

Таблица 36.22. Формат регистра gpio_ext_porta

Номер бита	Условное обозначение	Назначение
31:0	gpio_ext_porta	Если любой бит порта A настроен на прием данных (gpio_swporta_ddr[n] = 0), то чтение соответствующего бита регистра возвращает значение входного сигнала этого бита. Если бит порта A настроен на выдачу данных (gpio_swporta_ddr[n] = 1), то чтение этого бита значение возвращает значение бита gpio_swporta_dr[n].

36.2.22 Регистр gpio_ext_portb

Внешний регистр порта B.

Формат регистра gpio_ext_portb приведен в Таблица 36.23.

Таблица 36.23. Формат регистра gpio_ext_portb

Номер бита	Условное обозначение	Назначение
31:0	gpio_ext_portb	Если любой бит порта B настроен на прием данных (gpio_swportb_ddr[n] = 0), то чтение соответствующего бита регистра возвращает значение входного сигнала этого бита. Если бит порта B настроен на выдачу данных (gpio_swportb_ddr[n] = 1),

Номер бита	Условное обозначение	Назначение
		то чтение этого бита значение возвращает значение бита gpio_swportb_dr[n].

36.2.23 Регистр gpio_ext_portc

Внешний регистр порта C.

Формат регистра gpio_ext_portc приведен в Таблица 36.24.

Таблица 36.24. Формат регистра gpio_ext_portc

Номер бита	Условное обозначение	Назначение
31:0	gpio_ext_portc	Если любой бит порта C настроен на прием данных (gpio_swportc_ddr[n] = 0), то чтение соответствующего бита регистра возвращает значение входного сигнала этого бита. Если бит порта C настроен на выдачу данных (gpio_swportc_ddr[n] = 1), то чтение этого бита значение возвращает значение бита gpio_swportc_dr[n].

36.2.24 Регистр gpio_ext_portd

Внешний регистр порта D.

Формат регистра gpio_ext_portd приведен в Таблица 36.25.

Таблица 36.25. Формат регистра gpio_ext_portd

Номер бита	Условное обозначение	Назначение
31:0	gpio_ext_portd	Если любой бит порта D настроен на прием данных (gpio_swportd_ddr[n] = 0), то чтение соответствующего бита регистра возвращает значение входного сигнала этого бита. Если бит порта D настроен на выдачу данных (gpio_swportd_ddr[n] = 1), то чтение этого бита значение возвращает значение бита gpio_swportd_dr[n].

36.3 Функционирование блока управления GPIO

36.3.1 Программное и аппаратное управление

Когда какой-либо вывод настроен на программный источник данных (gpio_swport*x*_ctl=0), то направление потока данных регулируется через регистр gpio_swport*x*_ddr (0 - прием, 1 - выдача). Если вывод настроен на выдачу данных, то выходному сигналу передается значение регистра gpio_swport*x*_dr. В случае, если вывод настроен на прием данных, то значение принимаемого выводом сигнала отображается в регистр gpio_ext_port*x*.

Когда какой-либо вывод настроен на аппаратный источник данных(gpio_swportx_ctl=1), то назначение этого вывода определяется конфигурацией системы.

36.3.2 Прерывания

Любой бит порта A может воспринимать внешний сигнал как источник прерывания. Тип прерывания может быть следующим:

- активный высокий уровень;
- активный низкий уровень;
- передний фронт;
- задний фронт.

Прерывания могут быть маскированы через регистр gpio_intmask. Статус прерывания можно читать до маскирования через регистр gpio_raw_intstatus и после маскирования gpio_intstatus.

Чтобы настроить порт A на прерывания, необходимо установить в 1 регистр gpio_inten. Также необходимо, чтобы порт A был настроен на программный источник данных и на прием данных.

Для сброса прерывания по фронту необходимо записать 1 в соответствующий бит регистра gpio_porta_eoi. Это также сбросит значение соответствующего бита в регистре gpio_intstatus и gpio_raw_intstatus.

Для сброса прерывания по уровню необходимо устранить источник прерывания или маскировать его через регистр gpio_intmask.

Блок GPIO имеет функцию фильтрации дребезга входного сигнала, обрабатываемого как прерывание. При установке в 1 соответствующего бита регистра gpio_debounce, gpio сформирует прерывание, в случае если активный уровень входного сигнала длится 2 такта фильтрующей частоты gpio_dbclk.

37. КОНТРОЛЛЕР МНОГОФУНКЦИОНАЛЬНОГО БУФЕРИЗИРОВАННОГО ПОСЛЕДОВАТЕЛЬНОГО ПОРТА (MFBSP) С ПОДДЕРЖКОЙ CAN

37.1 MFBSP_PORT: MFBSP+CAN

Блок mfbsp_port объединяет в одном модуле контроллеры CAN и MFBSP. По выводам блок повторяет выводы блока mfbsp, которые мультиплексированы с выводами CAN.

37.1.1 Структурная схема MFBSP_PORT

На следующем рисунке показана структурная схема блока mfbsp_port:

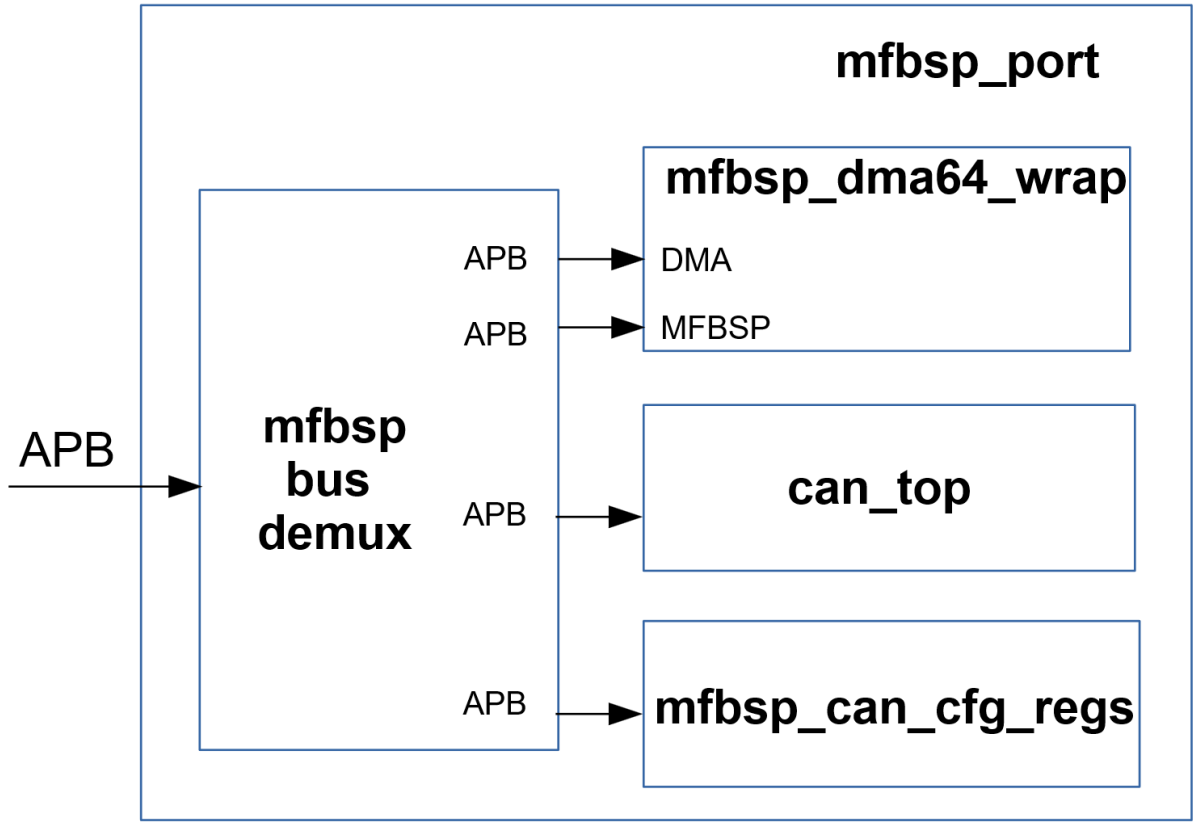


Рисунок 37.1. Структурная схема блока mfbsp_port

Входная шина APB разбивается на четыре, которые подключаются к блокам:

- mfbsp_dma64_wrap порт MFBSP
- mfbsp_dma64_wrap порт DMA
- can_top
- mfbsp_can_cfg_regs

Распределение памяти на шине APB:

Таблица 37.1. Распределение памяти на шине APB mfbsp_port

Адрес	Назначение
0x0000-0x12FF	Регистры контроллера CAN
0x1300-0x133F	Регистры контроллера MFBSP
0x1340-0x137F	Регистры DMA
0x1380-0x13BF	Регистры mfbsp_can_cfg_regs

Таблица 37.2. Регистры mfbsp_can_cfg_regs

Адрес	Назначение
0x00	Регистр CFG.
0x04-0x3F	Зарезервировано.

Таблица 37.3. mfbsp_can_cfg_regs.CFG

Разряды	Имя	Значение по сбросу	Назначение
31:1	–	0	Зарезервировано.
0	canenable	0	Разрешение работы контроллера CAN.

Выводы LDAT[7] и LDAT[6] совместно используются CAN контроллером и контроллером MFBSP. При установленном разряде canenable регистра mfbsp_can_cfg_regs.CFG к выходу LDAT[7] подключается выход TX_CAN контроллера CAN. Вывод LDAT[7] конфигурируется как выход, LDAT[6] как вход. LDAT[6] подключён ко входу контроллера CAN RX_CAN.

37.1.2 Мультиплексирование выводов mfbsp_port:

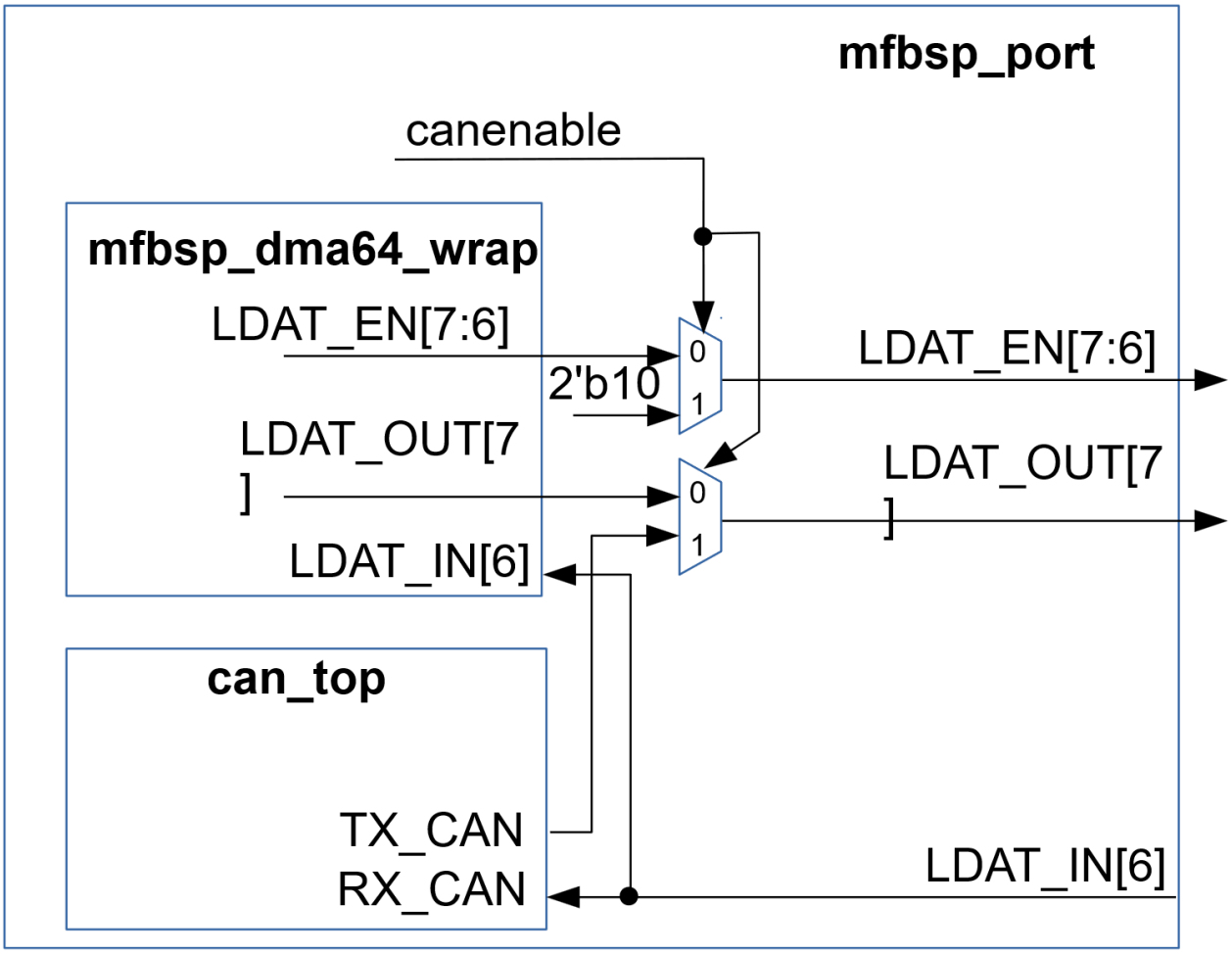


Рисунок 37.2. Мультиплексирование выводов mfbsp_port

37.2Особенности MFBSP

Многофункциональный буферизированный последовательный порт (MFBSP) позволяет вести обмен параллельно-последовательным кодом с другими микросхемами по линковому интерфейсу (LPORT), либо обмениваться аудиоданными и управляющей информацией с внешними устройствами по последовательным интерфейсам в дуплексном режиме, с возможностью независимой настройки приёмника и передатчика. Гибкость последовательного порта позволяет организовывать передачу с широким спектром внешних устройств. Дополнительно порт позволяет организовывать обмен данными с внешними устройствами, используя вводы-выводы общего назначения. На Рисунок 37.3 изображен MFBSP с двумя каналами DMA (на приём и передачу) в составе микропроцессора. По каналу DMA направления передачи осуществляется передача данных внешнему устройству, подключенному к микропроцессору через MFBSP. По каналу DMA направления приёма осуществляется приём данных из внешнего устройства, подключенного к микропроцессору через MFBSP.

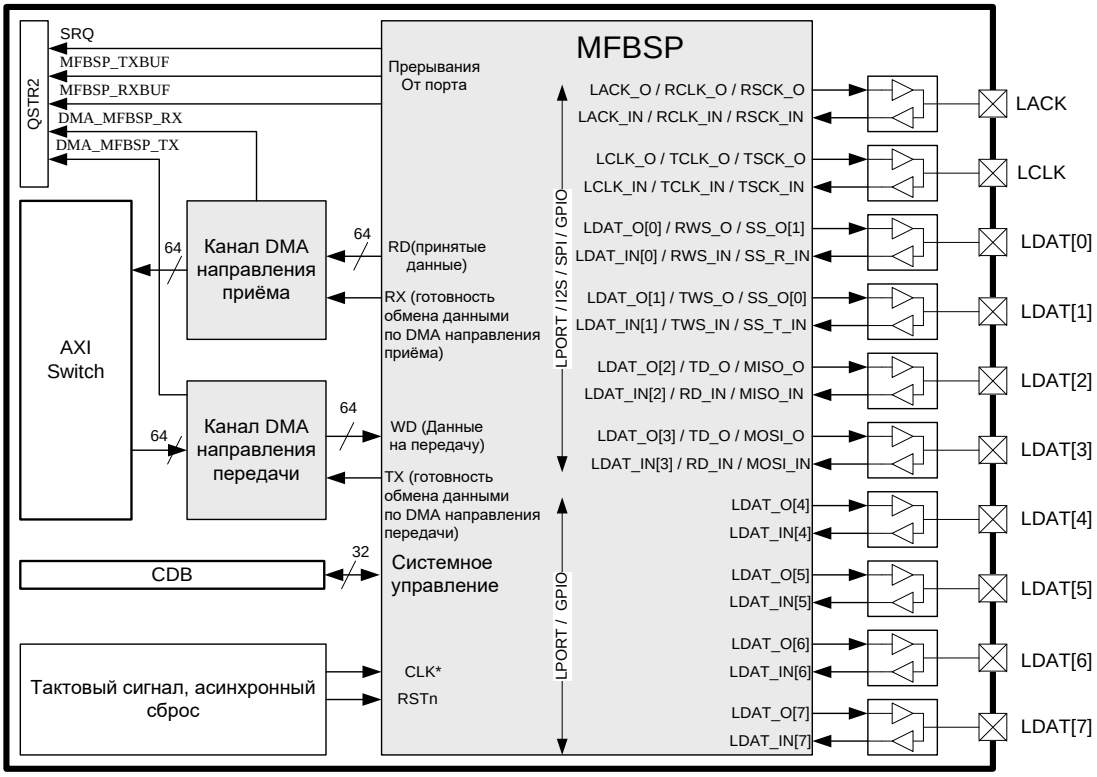


Рисунок 37.3. MFBSP в составе микропроцессора

37.2.1 Основные характеристики MFBSP в режиме I2S

В режиме I2S порт позволяет вести дуплексный обмен последовательными данными с внешними устройствами, используя следующие форматы передачи данных: Left-Justified, Right-Justified (при программной предобработке данных), DSP, I2S, FSB (Fast Serial Bus используемый в микросхеме СМХ981.

Ограничение использования формата I2S: приемник MFBSP, в текущей версии порта, в режиме ведомого устройства не позволяет принимать слова от передатчика, если число тактов между фронтами сигнала выбора канала меньше, чем RWORDLEN+1 (разрядность передаваемых слов меньше, чем установленная разрядность принимаемых передатчиком слов).

Приёмник и передатчик:

- поддерживается независимая настройка передатчика и приёмника, что позволяет организовать одновременно передачу и прием последовательных данных по разным последовательным интерфейсам и на различных частотах;
- возможен перевод приёмника в зависимый от передатчика режим (когда приемник использует тактовый и контрольный сигналы передатчика), что позволяет задействовать меньшее количество выводов;
- направление любого вывода задается программно, что заметно повышает гибкость при использовании порта;
- тактовые и управляющие сигналы как приемника, так и передатчика можно формировать аппаратными средствами порта MFBSP, либо принимать их от внешнего устройства.

Темп передачи данных:

- передача данных в режиме I2S может вестись на частотах от CLK/2 до CLK/(2*2^10) (где CLK – тактовая частота, подаваемая на порт со стороны системы);
- частоту контрольного сигнала (TWS/RWS) можно задавать в пределах от ICLK/2 до ICLK/(2*2^16), где ICLK – рабочая частота интерфейса (TCLK для передатчика и RCLK для приемника).

Приём и передача данных:

- порт позволяет принимать и передавать слова длиной от 2-х до 32-х бит, как младшим, так и старшим битом вперед;
- в режиме I2S поддерживается режим паковки/распаковки 32-х разрядного слова в два 16-ти разрядных с автоматическим определением левого/правого канала;
- специальная логика обмена позволяет обнулять или дополнять старшим разрядом избыточные биты при чтении принятых слов длиной меньше 32 в обычном режиме и длиной меньше 16 в режиме паковки;
- порт поддерживает приём и передачу данных фреймами с синхронизацией начала каждого фрейма. Число слов в одном фрейме может быть выбрано в пределах от 1 до 64.

Буферы приёма и передачи:

- используется буферизация в направлении передачи на 18 32-разрядных слов;

- используется буферизация в направлении приёма на 18 32-разрядных слов;
- доступ к буферам приёма и передачи возможен как в 32-х разрядном режиме (обмен данными непосредственно с CPU), так и в 64-х разрядном режиме с использованием каналов DMA;
- степень заполнения буфера передачи, при которой инициируется закачка данных в порт с помощью DMA, регулируется программно;
- для каждого порта MFBSP предусмотрено два независимых канала DMA на приём и на передачу.

37.2.2 Основные характеристики MFBSP в режиме SPI

В режиме SPI порт позволяет вести дуплексный обмен последовательными данными с внешними устройствами, порт поддерживает 4 формата передачи SPI (для всех сочетаний CPOL и CPHA по спецификации Motorola), при этом возможна передача данных как по стандарту Microwire (SDO, SDI), так и по стандарту Motorola (MOSI, MISO), а также по интерфейсу C-BUS (аналог SPI).

Приёмник и передатчик:

- поддерживается независимая настройка передатчика и приёмника, что позволяет организовать одновременные передачу и прием последовательных данных по разным последовательным интерфейсам и на различных частотах;
- возможен перевод приёмника в зависимый от передатчика режим (когда приемник использует тактовый и контрольный сигналы передатчика), что позволяет задействовать меньшее количество выводов;
- направление любого вывода задается программно, что заметно повышает гибкость при использовании порта.

Шина выбора ведомых устройств:

- тактовые сигналы и сигналы шины выбора ведомых устройств можно формировать аппаратными средствами порта MFBSP, программно управлять шиной выбора ведомых устройств, либо принимать тактовые сигналы и сигнал выбора ведомого от внешнего устройства;
- в режиме ведущего устройства портом используется двухразрядная шина выбора ведомых устройств, что позволяет параллельно подключать до двух ведомых SPI устройств;
- в режиме ведомого устройства возможен независимый выбор внешним устройством приёмника и передатчика MFBSP.

Темп передачи данных:

- передача данных в режиме SPI может вестись на частотах от CLK/2 до CLK/(2*2^10) (где CLK – тактовая частота, подаваемая на порт со стороны системы).

Приём и передача данных:

- порт позволяет принимать и передавать слова длиной от 2-х до 32-х бит, как младшим, так и старшим битом вперед;
- специальная логика обмена позволяет обнулять или дополнять старшим разрядом избыточные биты при чтении принятых слов длиной меньше 32 бит;
- порт позволяет вести обмен данными в режиме автоматического формирования сигналов выбора ведомого, с возможностью передачи от 1 до 64 слов без изменения уровня сигнала выбора ведомого.

Буферы приёма и передачи:

- используется буферизация в направлении передачи на 18 32-разрядных слов;
- используется буферизация в направлении приёма на 18 32-разрядных слов;
- доступ к буферам приёма и передачи возможен как в 32-х разрядном режиме (обмен данными непосредственно с CPU), так и в 64-х разрядном режиме с использованием каналов DMA;
- степень заполнения буфера передачи, при которой инициируется закачка данных в порт с помощью DMA, регулируется программно;
- для каждого порта MFBSP предусмотрено два независимых канала DMA на приём и на передачу;
- в данной реализации порта существует ограничение на выбор направления выводов в режиме SPI: тактовый и управляющий сигналы в режиме SPI должны быть либо оба заданы как вход, либо оба заданы как выход.

В данной реализации порта не предусмотрена возможность соединения нескольких микропроцессоров по цепочке с использованием SPI интерфейса. Микропроцессор может только управлять загрузкой последовательных данных в другие ведомые устройства, соединенные по цепочке.

37.2.3 Основные характеристики MFBSP в режиме LPORT

В режиме LPORT порт позволяет вести обмен с внешними устройствами по линковому интерфейсу (совместимому с ADSP21160 LINK PORT).

Приёмник и передатчик:

- В режиме LPORT MFBSP может работать либо только как передатчик, либо только как приёмник (передача данных в одном направлении);

Темп передачи данных:

- Передача данных по интерфейсу LPORT может вестись на частотах от CLK/32 до CLK/2 (где CLK – тактовая частота, подаваемая на порт со стороны системы);

Приём и передача данных:

- По параллельно-последовательному интерфейсу LPORT возможна передача данных как тетрадами, так и байтами;

Буферы приёма и передачи:

- используется буферизация в направлении передачи на 16 32-разрядных слов;
- используется буферизация в направлении приёма на 18 32-разрядных слов;
- доступ к буферам приёма и передачи возможен как в 32-х разрядном режиме (обмен данными непосредственно с CPU), так и в 64-х разрядном режиме с использованием каналов DMA;
- степень заполнения буфера передачи, при которой инициируется закачка данных в порт с помощью DMA, регулируется программно.

В зависимости от выбранного направления порта используется либо канал DMA направления приёма, либо канал DMA направления передачи.

37.2.4 Основные характеристики MFBSP в режиме порта ввода-вывода общего назначения

В режиме порта ввода-вывода общего назначения все 10 выводов порта могут использоваться как вводы выводы общего назначения.

Направление каждого вывода задаётся программно.

В режиме последовательного порта (режимы SPI или I2S) 4 незадействованных в передаче последовательных данных вывода MFBSP (LDAT[7:4]) могут быть использованы в качестве вводов-выводов общего назначения.

37.3Общие сведения об MFBSP

37.3.1 Режимы работы MFBSP

Многофункциональный порт MFBSP может быть использован как порт ввода-вывода общего назначения, как линковый порт (LPORT), либо как последовательный порт. В случае если MFBSP используется как последовательный порт, приёмник и передатчик могут настраиваться независимо. Как приёмник, так и передатчик MFBSP могут работать в режиме SPI либо в режиме I2S. Таким образом, для MFBSP существует 6 различных режимов работы, которые задаются битами LEN и SPI_I2S_EN регистра CSR_MFBSP, битом TMODE регистра TCTR и битом RMODE регистра RCTR. Режимы работы MFBSP и задающие их сочетания значений управляющих бит приведены в Таблица 37.4.

Таблица 37.4. Режимы работы MFBSP

Значение битов, задающих режим					Режим работы MFBSP
№	LEN	SPI_I2S_EN	TMODE	RMODE	
1	0	0	x	x	Порт ввода-вывода общего назначения
2	1	0	x	x	Линковый порт(LPORT)
3	0	1	0	0	Последовательный порт Передатчик – I2S Приёмник – I2S
4	0	1	0	1	Последовательный порт Передатчик – I2S Приёмник – SPI
5	0	1	1	0	Последовательный порт Передатчик – SPI Приёмник – I2S
6	0	1	1	1	Последовательный порт Передатчик – SPI Приёмник – SPI

Более подробное описание функциональных особенностей порта для режима I2S приведено в параграфе 37.4.

Более подробное описание функциональных особенностей порта для режима SPI приведено в параграфе 37.5.

Более подробное описание функциональных особенностей порта для режима LPORT приведено в параграфе 37.6.

Более подробное описание функциональных особенностей порта для режима порта ввода-вывода общего назначения приведено в параграфе 37.7.

37.3.2 Структурная схема многофункционального буферизированного последовательного порта

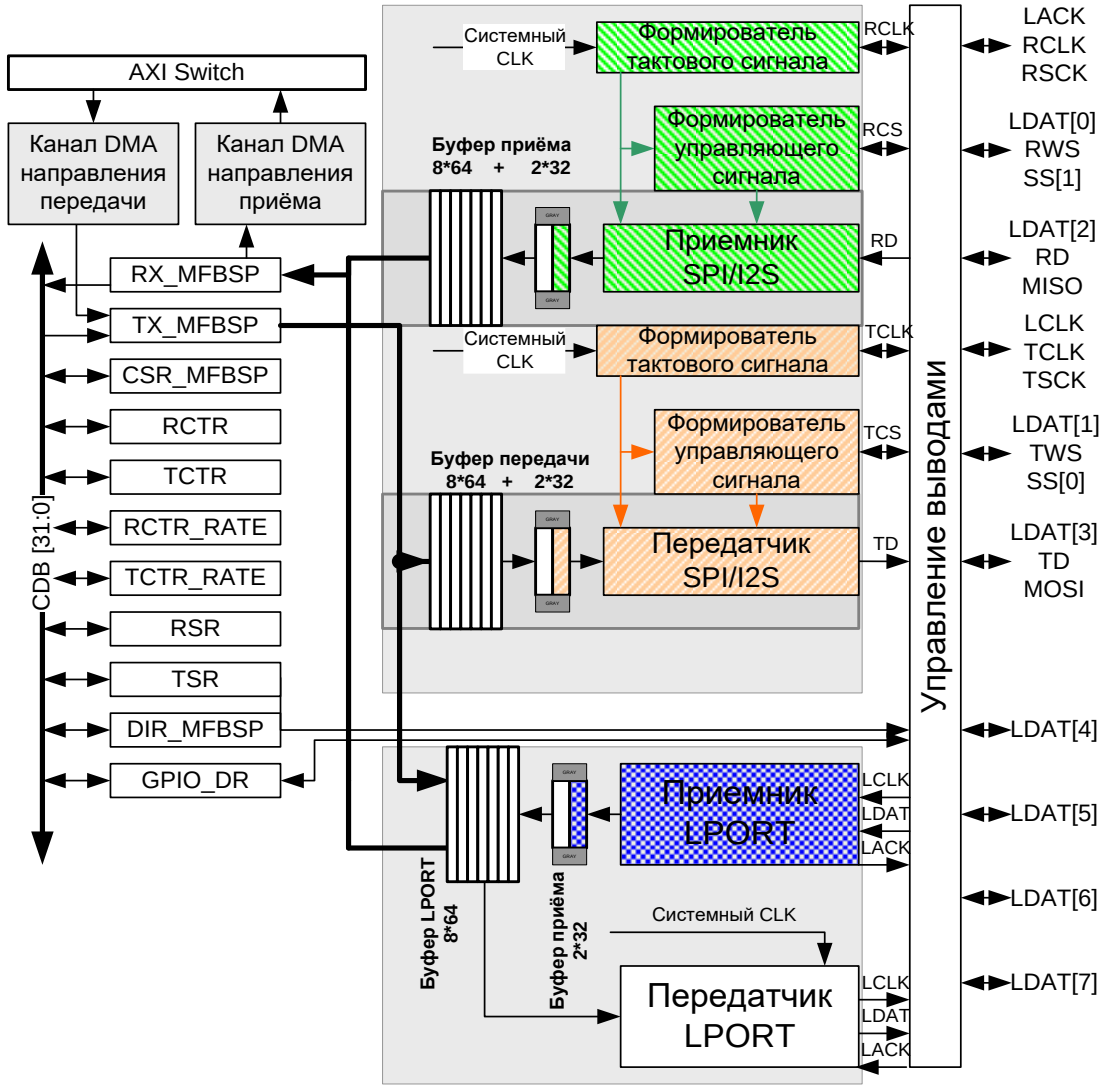


Рисунок 37.4. Структурная схема MFBSP (Защищена патентом РФ №2360282 от 27 июня 2009 года)

На Рисунок 37.3 показан MFBSP в составе микропроцессора. Порт поддерживает дуплексный обмен последовательными данными, поэтому для каждого MFBSP предусмотрено два канала DMA – один на приём и один на передачу. Каждый из внешних выводов порта двунаправленный, направление каждого вывода задается независимо.

На Рисунок 37.4 представлена более подробная структурная схема MFBSP.

В состав совмещенного контроллера входят два основных блока: ***контроллер LPORT*** и ***контроллер SPI/I2S***. Включение ***контроллера LPORT*** производится установкой бита LEN, регистра CSR_MFBSP в 1, включение ***контроллера SPI_I2S*** производится установкой бита SPI_I2S_EN, регистра CSR_MFBSP в 1 (Таблица 37.4). Одновременная работа блоков LPORT и SPI/I2S и соответственно установка бит LEN и SPI_I2S_EN в 1 не допускается.

В состав ***контроллера SPI/I2S*** входят приёмник, передатчик, буфер приёма и буфер передачи. Приёмник и передатчик могут работать одновременно и независимо. Приёмник осуществляет синхронный приём последовательного кода с внешнего вывода схемы и запись принятых данных в буфер приёма. Передатчик осуществляет чтение данных из буфера передачи и синхронную выдачу их последовательным кодом на внешний вывод схемы. Запись передаваемых данных в буфер передачи осуществляется при записи по адресу псевдорегистра TX_MFBSP (доступ со стороны CPU или DMA направления передачи), чтение принятых данных из буфера приёма осуществляется при чтении по адресу псевдорегистра RX_MFBSP (доступ со стороны CPU или DMA направления приёма).

Последовательным портом при обмене данными используется только 6 выводов LCLK, LACK, LDAT[3:0]. Если порт работает в режиме SPI/I2S, выводы LDAT[4:7] могут использоваться как вводы-выводы общего назначения.

В состав ***контроллера LPORT*** входят приёмник, передатчик и буфер LPORT. В зависимости от направления обмена данными работает либо приёмник, либо передатчик. В зависимости от направления обмена данными буфер LPORT выполняет функции либо буфера приёма, либо буфера передачи Приёмник осуществляет синхронный приём параллельно-последовательного кода с внешних выводов схемы и запись принятых данных в буфер LPORT. Передатчик осуществляет чтение данных из буфера LPORT и синхронную выдачу их параллельно последовательным кодом на внешние выводы схемы. Запись передаваемых данных в буфер LPORT осуществляется при записи по адресу псевдорегистра TX_MFBSP (доступ со стороны CPU или DMA направления передачи), чтение принятых данных из буфера LPORT осуществляется при чтении по адресу псевдорегистра RX_MFBSP (доступ со стороны CPU или DMA направления приёма).

LPORT при обмене данными использует выводы LCLK, LACK, LDAT[7:0].

При отключенной частоте MFBSP чтение и запись в регистры MFBSP0, MFBSP1 не допускается.

37.3.3 Назначение выводов порта в различных режимах

Таблица 37.5 содержит наименования выводов порта для каждого из режимов – LPORT, SPI, I2S. Таблица 37.6 содержит информацию о назначении каждого вывода в различных режимах.

Таблица 37.5. Обозначение выводов порта для различных режимов работы

LPORT	I2S	SPI
LDAT[7]	-	-
LDAT[6]	-	-
LDAT[5]	-	-
LDAT[4]	-	-
LDAT[3]	TD	MOSI

LDAT[2]	RD	MISO
LDAT[1]	TWS	SS[0]
LDAT[0]	RWS	SS[1]
LCLK	TCLK	TSCK
LACK	RCLK	RSCK

Таблица 37.6. Назначение выводов порта в различных режимах

Наименование вывода	Режим работы порта	Направление вывода	Назначение вывода
LDAT[7:0]	LPORT	IO	Внешняя шина данных LPORT.
LCLK	LPORT	IO	Тактовый сигнал LPORT
LACK	LPORT	IO	Подтверждение готовности приема
TD	I2S	IO	Передаваемые последовательные данные
RD	I2S	IO	Принимаемые последовательные данные
TCLK	I2S	IO	Тактовый сигнал передатчика I2S
RCLK	I2S	IO	Тактовый сигнал приемника I2S
TWS	I2S	IO	Сигнал выбора канала для передаваемых данных
RWS	I2S	IO	Сигнал выбора канала для принимаемых данных
MOSI	SPI	IO	Вывод последовательных данных. Направление вывода определяется программно
MISO	SPI	IO	Вывод последовательных данных. Направление вывода определяется программно
TSCK	SPI	IO	Тактовый сигнал передатчика SPI
RSCK	SPI	IO	Тактовый сигнал приемника SPI
SS [0]	SPI	IO	В режиме ведущего: Сигнал выбора устройства 0. В режиме ведомого: сигнал выбора ведомого. Низкий уровень на входе SS[0] обозначает, что передатчику MFBSP необходимо выдавать последовательные данные (если приёмник MFBSP находится в зависимом от передатчика режиме, то активизируется и приёмник).
SS [1]	SPI	IO	В режиме ведущего: Если приёмник в зависимом от передатчика режиме - сигнал выбора устройства 1. Если передатчик в независимом от приёмника режиме – сигнал выбора приёмником устройства 0. В режиме ведомого: Сигнал выбора ведомого. Только в случае когда приёмник в независимом от передатчика режиме. Низкий уровень на входе SS[1] обозначает, что приёмнику MFBSP необходимо принимать последовательные данные.

37.3.4 Перечень регистров MFBSP

Таблица 37.7 содержит перечень регистров многофункционального порта.

Таблица 37.7. Перечень регистров многофункционального буферизированного порта

Условное обозначение регистра	Смещение	Доступ	Название регистра
TX_MFBSP	0x0	W	Буфер передачи данных
RX_MFBSP	0x0	R	Буфер приёма данных
CSR_MFBSP	0x4	R/W	Регистр управления и состояния
DIR_MFBSP	0x8	R/W	Регистр управления направлением выводов порта ввода-вывода
GPIO_DR	0xC	R/W	Регистр данных порта ввода-вывода
TCTR	0x10	R/W	Регистр управления передатчиком
RCTR	0x14	R/W	Регистр управления приёмником
TSR	0x18	R/W	Регистр состояния передатчика
RSR	0x1C	R/W	Регистр состояния приёмника
TCTR_RATE	0x20	R/W	Регистр управления темпом передачи данных
RCTR_RATE	0x24	R/W	Регистр управления темпом приёма данных
TSTART	0x28	R/W	псевдорегистр ten – запуск/останов передатчика без изменения настроек передатчика
RSTART	0x2C	R/W	псевдорегистр ren – запуск/останов приемника без изменения настроек приемника
EMERG_MFBSP	0x30	R/W	Регистр аварийного управления портом
IMASK_MFBSP	0x34	R/W	Регистр маски прерываний от порта

37.3.5 Каналы DMA многофункциональных портов MFBSP

Для каждого порта предусмотрено два канала DMA – один для приема данных, другой для передачи данных.

По каналу DMA направления передачи осуществляется передача данных внешнему устройству, подключенному к микропроцессору через MFBSP. По каналу DMA направления приёма осуществляется приём данных из внешнего устройства, подключенного к микропроцессору через MFBSP.

При обмене данными через MFBSP с использованием DMA максимальный размер пачки составляет 8 64-разрядных слов. Если значение бит WN в контрольном регистре DMA превосходит максимальный размер пачки, то WN автоматически корректируется (большая пачка передается в несколько этапов пачками меньшего размера).

По умолчанию при работе передатчика с DMA заполнение буфера передачи происходит до тех пор, пока буфер в состоянии принять очередную пачку, размером WN. Однако имеется возможность программно регулировать степень заполнения буфера передачи, путем установки бит TBES, регистра TSR. В этом случае значение выражения TBES+1 – задает эффективный размер буфера передачи. Передача очередной пачки происходит только в случае, если при записи этой пачки в буфер передачи число 64 разрядных слов в буфере передачи не превысит TBES+1. При попытке передать пачку со значением WN > TBES, значение WN автоматически корректируется (большая пачка передается в несколько этапов пачками меньшего размера).

По умолчанию при работе приёмника с DMA, считывание данных из буфера приёма происходит, если в буфере чтения содержится число слов большее, либо равное размеру

пачки (WN). Степень заполнения буфера приёма, при которой начинается откачка данных с помощью DMA, регулируется установкой значения WN соответствующего канала DMA.

37.3.5.1 Каналы DMA периферийных портов

Для обслуживания порта MFBSP имеются следующие каналы DMA:

MFBSP _TX_CH1, MFBSP _TX_CH0,

MFBSP _RX_CH1, MFBSP _RX_CH0.

Для управления работой каналы DMA портов содержат следующие регистры:

- регистр управления и состояния (CSR);
- регистр индекса (физический адрес памяти) (IR);
- регистр начального адреса блока параметров DMA передачи для самоинициализации (CP);
- псевдорегистр управления состоянием бита RUN регистра CSR.

Исходное состояние регистров CSR: разряды 15:0 – нули, а состояние разрядов 31:16 не определено. Исходное состояние остальных регистров не определено.

Каналы DMA портов передают данные по коммутатору AXI Switch 64-разрядными словами.

32-разрядный индексный регистр IR содержат физический адрес внутренней или внешней памяти. Этот адрес должен быть выровнен по границе 64-разрядного слова. После передачи каждого слова данных к индексу IR прибавляется смещение на одно 64-х разрядное слово.

Памятью могут быть CRAM, блоки памяти DSP: XRAM, YRAM и PRAM, внешняя память, доступная через GPMC, DDRMC.

Формат регистров управления и состояния CSR каналов DMA этих портов приведен в Таблица 37.8.

Таблица 37.8. Формат регистров управления и состояния DMA портов

Номер разряда	Условное обозначение	Назначение
0	RUN	Состояние работы канала DMA: 0 – состояние останова; 1 – состояние обмена данными. Устанавливается в 1 при записи 1 в этот разряд. Устанавливается в 0: при завершении передачи блока данных. Состояние этого бита определяется в процессе выполнения процедуры самоинициализации
1	-	Резерв
5:2	WN	Число слов данных (пачка), которое передается за одно предоставление прямого доступа: 0 – 1 слово, F – 16 слов. Посредством этого параметра можно плавно изменять приоритет каналов DMA относительно других устройств и относительно друг друга
11:6	-	Резерв
12	CHEN	Разрешение выполнения очередной процедуры самоинициализации: 0 – выполнение очередной процедуры самоинициализации запрещено; 1 – выполнение очередной процедуры самоинициализации разрешено. Используется только при обмене цепочкой блоков данных. Состояние этого бита определяется в процессе выполнения процедуры самоинициализации
13	IM	Маска разрешение установки признака END: 0 – установки признака запрещено; 1 – установки признака разрешено. Используется только при обмене цепочкой блоков данных. Состояние этого бита определяется в процессе выполнения процедуры самоинициализации
14	END	Признак завершения передачи блока данных. Устанавливается в 1 при завершении передачи блока данных (при IM=1). Устанавливается в 0 при чтении содержимого этого регистра.
15	DONE	Признак завершения передачи блока данных. Устанавливается в 1 при завершении передачи блока данных при CHEN=0 (CHEN=1 может быть только при использовании процедуры самоинициализации). Устанавливается в 0 при чтении содержимого этого регистра
31:16	WCX	Число 64-разрядных слов данных, которые должен передать канал DMA (блок данных); количество передаваемых слов: WCX + 1;. содержимое этого поля уменьшается на 1 после передачи каналом DMA очередного слова данных.

Все разряды регистра CSR доступны по записи и чтению.

Бит RUN может быть использован для остановки работы канала DMA портов. Для этого в любой момент времени в него необходимо записать 0. Эта процедура возможна, если длина массива данных, указанного в канале DMA порта, равна длине массива данных, который порт передаст (например, MFBSP). Для продолжения работы в бит RUN необходимо записать 1.

Если MFBSP прекратил обмен данными по внешней причине, то длина массива данных, указанного в канале DMA порта, будет не равна длине массива данных, который порт действительно передаст. В этом случае для остановки работы порта и его канала DMA необходимо использовать следующий алгоритм.

Алгоритм остановки MFBSP и его канала DMA:

- Остановить MFBSP, для чего в регистр CSR_MFBSP необходимо записать 0. Данная запись остановит и приемник и передатчик, если нужно остановить только приемник или только передатчик (для режимов SPI и I2S) необходимо использовать запись 0 в бит REN, регистра RCTR для останова приемника, или TEN, регистра TCTR для останова передатчика.

- Выполнить операцию записи 0 в бит RUN регистра CSR соответствующего канала DMA MFBSP (при этом, бит RUN может в 0 не установиться).

- Установить в регистре EMERG_MFBSP в 1 бит RX_DBG, если останавливался приемник или TX_DBG, если останавливался передатчик.

- Дождаться установки в 0 бита RUN регистра CSR соответствующего канала DMA MFBSP.

- Установить в регистре EMERG_MFBSP в 0 биты RX_DBG и TX_DBG.

- Для режимов SPI или I2S: сбросить буфер приема или буфер передачи (в зависимости от того останавливался приемник или передатчик) записью 1 в бит RST_RXBUF или RST_TXBUF, регистра EMERG_MFBSP. Для режима LPORT: сбросить буфера записью 1 в бит RST_LPTBUF.

Следует отметить, что при выполнении этого алгоритма «хвост» передаваемых данных из порта теряется, а в «хвосте» приемного буфера данные будут недостоверны.

Состоянием разряда 0 регистра CSR можно управлять, используя адрес псевдорегистра RUN. При этом остальные разряды этого регистра не изменяются. Эта процедура может быть использована для временной приостановки канала DMA. При чтении по адресу псевдорегистра RUN считывается содержимое регистра CSR без сброса битов END и DONE.

37.3.6 Процедура самоинициализации

Все каналы DMA могут выполнять процедуру самоинициализации (выполнение цепочки передач DMA).

Для выполнения самоинициализации в каналах DMA имеется 32-разрядный регистр CP, в котором хранится физический начальный адрес блока параметров очередного DMA обмена. Младшие три разряда регистра CP игнорируются (адреса выровнены по границе 64-разрядного слова). Младший (нулевой разряд) регистра CP используется для старта режима самоинициализации. Эти параметры при самоинициализации аппаратно загружаются в 64-разрядном формате в соответствующие регистры канала DMA. Процедура этой загрузки ничем не отличается от обычного DMA обмена. Блок параметров может размещаться в любой памяти микросхемы.

Если необходимо продолжить цепочку команд, то необходимо указать CHEN=1. В режиме самоинициализации при записи параметров в регистр CSR биты END и DONE недоступны.

Для запуска работы канала DMA в режиме с самоинициализацией необходимо в регистр CP записать адрес первого блока параметров DMA передачи. При этом 0 разряд записываемых данных должен содержать 1 (признак пуска самоинициализации). В результате этого, соответствующий канал загрузит в свои регистры параметры DMA передачи и начнет обмен данными.

После окончания передачи блока данных бит END в регистре CSR устанавливается в единичное состояние, если бит IM = 1 - выдается прерывание. По окончании передачи блока данных также проверяется состояние бита CHEN. Если он равен 1, то будет загружен следующий блок параметров DMA передачи и т.д. В противном случае цепочка DMA обменов закончится и в регистре CSR бит DONE установится в единичное состояние и выдается прерывание.

Параметры для самоинициализации каналов DMA MEM_CH размещаются в памяти в трех последовательных 64-разрядных словах, следующим образом (в порядке возрастания адресов):

63_____0
{ $IR1_{32}$, $IR0_{32}$ };
{{WCY_{16},ORY_{16}},{ $OR1_{16}$,$OR0_{16}$ }};
{ CSR_{32}, CP_{32} }.

Параметры для самоинициализации каналов DMA портов размещаются в памяти в двух последовательных 64-разрядных словах, следующим образом (в порядке возрастания адресов):

63_____0

```
{ IR32,    -32 };

{ CSR32, CP32 } .
```

При необходимости каналы DMA могут инициализироваться программно. Для этого CPU должен загрузить все необходимые регистры индекса и смещения, а затем регистр CSR. При загрузке регистра CSR бит RUN необходимо установить в единичное состояние. Следует отметить, что бит RUN может быть использован для приостановки канала DMA. Для этого в любой момент времени в него необходимо записать 0. Для продолжения работы соответственно в бит RUN необходимо записать 1. Бит RUN может быть использован также для приостановки выполнения цепочки, если при загрузке очередных параметров он будет равен 0. Для продолжения выполнения цепочки в бит RUN необходимо записать 1. Для удобства организации обмена только с битом RUN имеется специальный регистр.

37.3.7 Прерывания от каналов DMA MFBSP

Бит DMA_MFBSP_RX, регистра QSTR2, устанавливается, если есть прерывание от соответствующего порту канала DMA направления приёма.

Бит DMA_MFBSP_TX, регистра QSTR2, устанавливается, если есть прерывание от соответствующего порту канала DMA направления передачи.

Если соответствующий канал DMA разрешен, то прерывания от канала DMA формируются по завершению передачи или приема всего блока данных.

37.3.8 Прерывания от MFBSP

Бит MFBSP_TXBUF, регистра QSTR2, устанавливается в случае, если число 64-х разрядных слов, находящихся в буфере передачи, меньше, либо равно пороговому значению TLEV, задаваемому в регистре TSR (Рисунок 37.5). Для установки бита MFBSP_TXBUF также необходимо, чтобы линковый порт был включен на передачу (LEN=1 и LTRAN=1) либо включен передатчик SPI/I2S (SPI_I2S_EN=1, TEN=1) и разрешена установка прерывания MFBSP_TXBUF по условию превышения порога (TX_LEV_IRQ_EN). MFBSP_TXBUF также устанавливается в случае, если имела место ошибка передачи TERR и разрешена установка прерывания MFBSP_TXBUF при ошибке передачи (чтение из пустого буфера передачи) : TX_ERR_IRQ_EN.

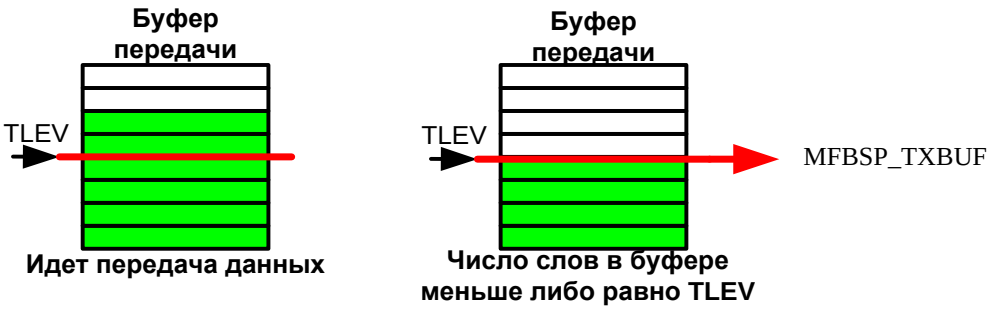


Рисунок 37.5. Назначение бит TLEV, регистра TSR

MFBSP_TXBUF может формироваться в двух режимах: с автоматическим сбросом при чтении регистра TSR и без автоматического сброса при чтении TSR. Данный режим определяется состоянием бита TXBUF_R_EN, регистра IMASK_MFBSP.

В случае если выбран режим без автоматического сброса, прерывание установлено всегда, когда имеет место превышение уровня TLEV. В случае если установлен режим с автоматическим сбросом, управление битом MFBSP_TXBUF осуществляется следующим образом: прерывание MFBSP_TXBUF автоматически сбрасывается, если число 64-х разрядных слов, находящихся в буфере передачи, становится больше порогового значения TLEV и при этом во время передачи не возникало ошибки (TERR = 0). Даже если описанное условие не выполнено, прерывание можно программно сбросить, прочитав регистр TSR. В этом случае прерывание сбросится и запомнится текущее значение слов в буфере передачи. Если число слов в буфере передачи начнет уменьшаться или произойдет ошибка передачи, то прерывание снова установится. Увеличение числа слов в буфере передачи не приведет к установке прерывания, даже, если число слов в буфере ниже порога TLEV (Рисунок 37.6).

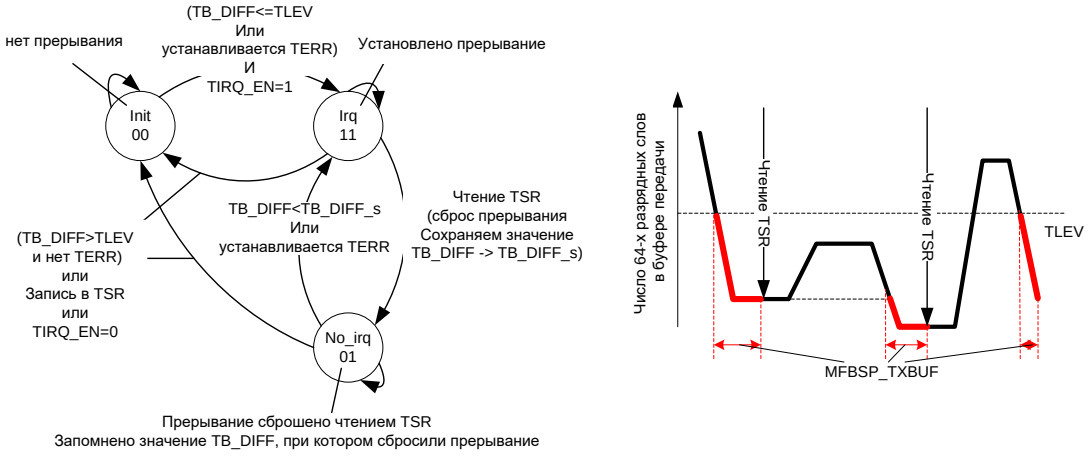


Рисунок 37.6. Механизм установки и сброса прерывания MFBSP_TXBUF. На рисунке TIRQ_EN = (LEN & LTRAN || TEN & SPI_I2S_EN)

Бит MFBSP_RXBUF, регистра QSTR2, устанавливается в случае, если число 64-х разрядных слов в буфере приёма больше чем пороговое значение RLEV, задаваемое в регистре RSR (Рисунок 37.7). Для установки бита MFBSP_RXBUF также необходимо, чтобы линковый порт был включен на приём (LEN=1 и LTRAN=0) либо включен приёмник SPI/I2S (SPI_I2S_EN=1, REN=1) и разрешена установка прерывания MFBSP_RXBUF по

условию превышения порога прерывания (TX_LEV_IRQ_EN). MFBSP_RXBUF также устанавливается в случае, если имела место ошибка приема RERR и разрешена установка прерывания MFBSP_RXBUF при ошибке передачи (запись в полный буфер приема) : RX_ERR_IRQ_EN.

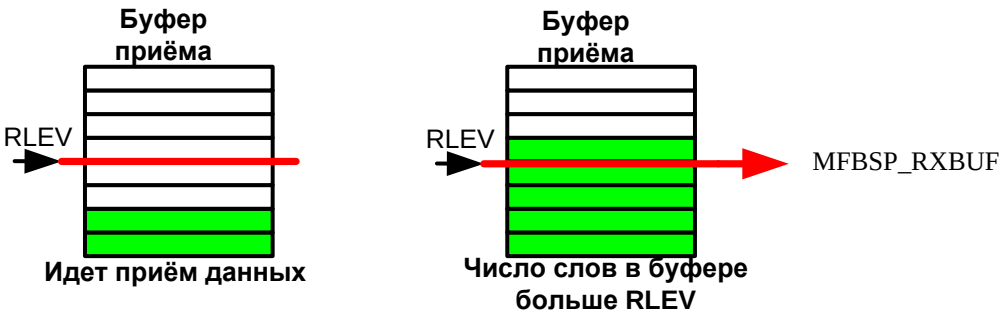


Рисунок 37.7. Назначение бит RLEV, регистра RSR

MFBSP_RXBUF может формироваться в двух режимах: с автоматическим сбросом при чтении регистра RSR и без автоматического сброса при чтении RSR. Данный режим определяется состоянием бита RXBUF_R_EN, регистра IMASK_MFBSP.

В случае если выбран режим без автоматического сброса, прерывание установлено всегда, когда имеет место превышение уровня RLEV. В случае если установлен режим с автоматическим сбросом, управление битом MFBSP_RXBUF осуществляется следующим образом: прерывание MFBSP_RXBUF автоматически сбрасывается, если число 64-х разрядных слов, находящихся в буфере приёма, становится меньше порогового значения RLEV и при этом во время приёма не возникало ошибки (RERR = 0). Даже если описанное условие не выполнено, прерывание можно программно сбросить, прочитав регистр RSR. В этом случае прерывание сбросится и запомнится текущее значение слов в буфере чтения. Если число слов в буфере чтения начнет увеличиваться, то прерывание снова установится. Уменьшение числа слов в буфере чтения не приведет к установке прерывания, даже, если превышен порог RLEV (Рисунок 37.8).

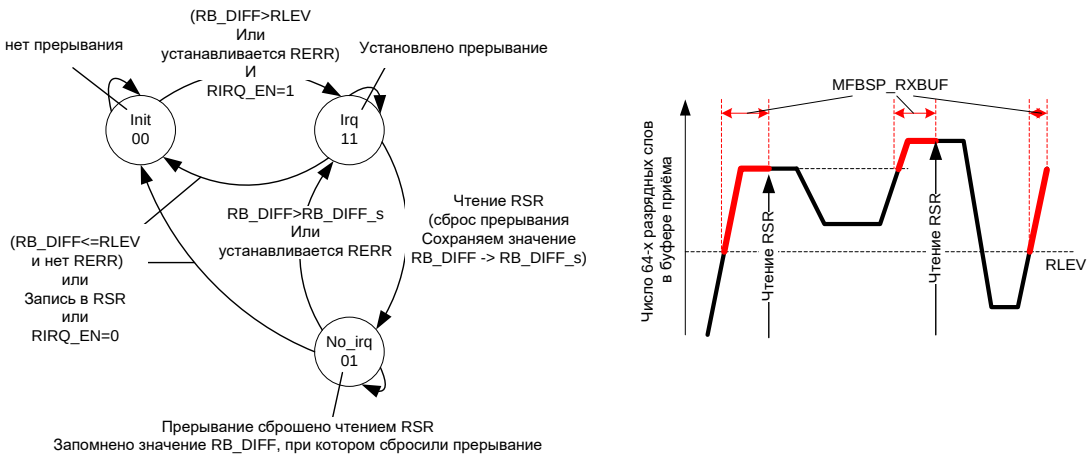


Рисунок 37.8. Механизм установки и сброса прерывания MFBSP_RXBUF. На рисунке RIRQ_EN = (LEN & !LTRAN || REN & SPI_I2S_EN)

Бит SRQ, регистра QSTR2, формируется при запросе на обслуживание, если порт MFBSP выключен (LEN=0, SPI_I2S_EN=0) и на выводах LACK или LCLK высокий уровень, при условии, что разрешено прерывание по запросу на обслуживание (LPT_IRQ_EN=1).

37.4 Работа MFBSP в режиме I2S

37.4.1 Назначение MFBSP в режиме I2S

Режим I2S буферизированного последовательного порта предназначен для организации дуплексного обмена аудиоданными с внешними устройствами последовательным кодом.

Порт в режиме I2S позволяет одновременно передавать и принимать последовательные данные. Приемник и передатчик контроллера настраиваются независимо, при этом возможен перевод приёмника в зависимое от передатчика состояние.

Порт поддерживает передачу аудиоданных в формате I2S, с поочередной передачей левого и правого каналов, а также передачу данных фреймами от 1 до 64 слов в каждом фрейме.

Поддерживается независимое задание направления каждого из выводов порта, осуществляемое установкой соответствующих бит регистра DIR_MFBSP.

Ограничение использования формата I2S: приемник MFBSP, в текущей версии порта, в режиме ведомого устройства не позволяет принимать слова от передатчика, если число тактов между фронтами сигнала выбора канала меньше, чем RWORDLEN+1 (разрядность передаваемых слов меньше, чем установленная разрядность принимаемых передатчиком слов).

37.4.2 Регистр управления и состояния CSR_MFBSP (режим I2S)

Регистр CSR_MFBSP (Таблица 37.9) используется для включения режима последовательного порта и разрешения прерываний от MFBSP.

Таблица 37.9. Назначение разрядов регистра CSR_MFBSP в режиме I2S

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:15	-	Резерв	-	0
14:11	-	В режиме I2S не используется	-	0
10	-	Резерв	-	0
9	SPI_I2S_EN	Включение режима SPI/I2S: 0 – Работа в режиме LPORT 1 – Работа в режиме SPI/I2S	RW	0
8:5	-	В режиме I2S не используется	-	0
4:3	LSTAT	Состояние буфера: При LTRAN = 0 показывает состояние буфера приёма При LTRAN = 1 показывает состояние буфера передачи 00 – буфер пуст; 10 – буфер не пуст; 11 – буфер полон.	R	0
2	-	В режиме I2S не используется	-	0
1	LTRAN	Назначение бит LSTAT:	RW	0

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
		0 - LSTAT отображает состояние буфера приёма 1 - LSTAT отображает состояние буфера передачи		
0	LEN	В режиме I2S должен быть установлен в 0	RW	0

37.4.3 Регистр управления направлением выводов DIR_MFBSP (режим I2S)

Регистр управления направлением выводов DIR_MFBSP (Таблица 37.10) предназначен для индивидуальной настройки направления каждого вывода последовательного порта.

Таблица 37.10. Назначение разрядов регистра DIR_MFBSP в режиме I2S

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
9:6	LDAT_DIR[7:4]	Направление выводов LDAT[7:4]	RW	0
5	TD_DIR	Направление вывода TD: 0 – TD – вход (при RD_DIR = 1 последовательные данные принимаются со входа TD) 1 – TD – выход (TD – является выходом для передачи последовательных данных)	RW	0
4	RD_DIR	Направление вывода RD: 0 – RD – вход (последовательные данные принимаются со входа RD) 1 – RD – выход (RD – является выходом для передачи последовательных данных)	RW	0
3	TCS_DIR	Направление вывода TWS: 0 – TWS – вход (Сигнал выбора слова TWS принимается от внешнего источника) 1 – TWS – выход (Сигнал выбора слова TWS формируется передатчиком)	RW	0
2	RCS_DIR	Направление вывода RWS: 0 – RWS – вход (Сигнал выбора слова RWS принимается от внешнего источника) 1 – RWS – выход (Сигнал выбора слова RWS формируется приёмником)	RW	0
1	TCLK_DIR	Направление вывода TCLK: 0 – TCLK – вход (тактовый сигнал TCLK принимается от внешнего источника) 1 – TCLK – выход (тактовый сигнал TCLK формируется передатчиком)	RW	0
0	RCLK_DIR	Направление вывода RCLK: 0 – RCLK – вход (тактовый сигнал RCLK принимается от внешнего источника) 1 – RCLK – выход (тактовый сигнал RCLK формируется приёмником)	RW	0

Примечания:

При RD_DIR = 0 и TD_DIR = 0 данные снимаются с RD.

При RD_DIR = 1 и TD_DIR = 1 на TD и RD выдаются одинаковые данные с передатчика.

37.4.4 Регистр управления приёмником RCTR (режим I2S)

Таблица 37.11. Назначение разрядов регистра RCTR в режиме I2S

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:30	-	Резерв	-	0
29	RCS_CONT	Включение непрерывного формирования сигнала RWS: 0 – RWS – Формируется если буфер приёма не полон. По заполнении буфера приёма формирование сигнала RWS прекращается. 1 – RWS – формируется непрерывно, если установлен бит REN	RW	0
28	RCLK_CONT	Включение непрерывного формирования сигнала RCLK: 0 – RCLK – формируется только во время приема (пока буфер приёма не полон). Если буфер приёма полон – сигнал не формируется 1 – RCLK – формируется непрерывно, если установлен бит REN	RW	0
27	RSWAP	Порядок упаковки в 32 разрядное слово, перед записью в буфер приёма: 0 – левый канал пишется в старшие 16 разрядов 1 – левый канал пишется в младшие 16 разрядов (Используется в режиме с включенным паковщиком)	RW	0
26	RSIGN	Значение заполнителя: Если длина принимаемого слова меньше 32 при отключенном паковщике или меньше 16 при включенном паковщике, то неиспользуемые биты принятого слова заполняются При RSIGN = 0 нулями При RSIGN = 1 значением старшего разряда в принятом слове	RW	0
25	RPACK	Включение режима паковки: 0 – режим паковки выключен. Данные, принятые по каждому из каналов пишутся отдельным 32-разрядным словом в буфер приёма 1 – режим паковки включен. Данные, принятые по левому и правому каналу пакуются в 32-х разрядное слово. При этом разрядность принимаемых слов не должна превышать 16.	RW	0
24:20	RWORDLEN	Длина принимаемого слова: Число бит в принимаемом слове равно RWORDLEN + 1. RWORDLEN должно быть больше 0.	RW	5’b0
19	RMBF	Порядок передачи бит: 0 – младшим битом вперед 1 – старшим битом вперед	RW RW	1
18	RCSNEG	Полярность управляющего сигнала приёмника: При RDSPMODE=0: RCSNEG = 0 –левый канал принимается при высоком уровне RWS RCSNEG = 1 – левый канал принимается при низком уровне RWS каждый фронт контрольного сигнала является активным и инициирует приём нового слова. При RDSPMODE=1: задаёт полярность активного фронта: RCSNEG = 0 - передний фронт активный; RCSNEG = 1 - задний фронт активный;	RW	0

17:12	RWORDCNT	Число слов во фрейме: Определяет число принимаемых в течении одного фрейма слов. Число принимаемых слов равно RWORDCNT + 1. Число бит, принимаемых в пределах одного фрейма, равно (RWORDCNT + 1)*(RWORDLEN+1) При RPACK = 1 обязательно RWORDCNT = 0	RW	0
11	RDEL	Задержка начала приёма данных на такт: 0 – захват бит принимаемого слова начинается по первому после активного фронта управляющего сигнала RWS фронту приёма такового сигнала RCLK (используется для передачи в форматах Left-Justified и Right-Justified) 1 – захват бит принимаемого слова начинается по второму после активного фронта управляющего сигнала RWS фронту приёма такового сигнала RCLK (используется для передачи в формате I2S)	RW	0
10	RNEG	Полярность тактового сигнала приёмника: Задает исходное состояние вывода RCLK и фронт, по которому осуществляется захват данных приёмником (фронт приёма) 0 – захват данных по заднему фронту RCLK. 1 – захват данных по переднему фронту RCLK. Исходное состояние RCLK = RNEG.	RW	0
9	RDSPMODE	Формат передачи данных: 0 – передача в формате I2S 1 – передача в формате DSP	RW	0
8:4	-	Резерв	-	0
3	RCS_CP	Дублирование сигнала TWS: 0 – выводы TWS и RWS независимы 1 – сигнал RWS, идущий на блок приёмника, дублирует TWS	RW	0
2	RCLK_CP	Дублирование TCLK: 0 – выводы TCLK и RCLK независимы 1 – сигнал RCLK, идущий на блок приёмника, дублирует TCLK	RW	0
1	RMODE	Режим работы приёмника: 0 – режим I2S 1 – режим SPI	RW	0
0	REN	Разрешение работы приёмника: 0 – приемник выключен 1 – приемник включен	RW	0

37.4.5 Регистр управления передатчиком TCTR (режим I2S)

Таблица 37.12. Назначение разрядов регистра TCTR в режиме I2S

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:30	-	В режиме I2S не используется	-	0
29	TCS_CONT	Включение непрерывного формирования сигнала TWS: 0 – TWS – формируется только если буфер передачи не пуст. После передачи последнего слова из буфера передачи формирование сигнала TWS прекращается 1 – TWS – формируется непрерывно, если установлен бит TEN	RW	0

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
28	TCLK_CONT	Включение непрерывного формирования сигнала TCLK: 0 – TCLK – формируется только во время передачи. Если буфер передачи пуст – сигнал не формируется 1 – TCLK – формируется непрерывно, если установлен бит TEN	RW	0
27	TSWAP	Порядок распаковки 32-х разрядного слова: Определяет порядок распаковки из 32 разрядного слова 0 – в левый канал передаются старшие 16 разрядов 1 – в левый канал передаются младшие 16 разрядов (Используется в режиме с включенным распаковщиком)	RW	0
26	-	Резерв	-	0
25	TPACK	Включение режима распаковки: 0 – режим распаковки выключен. Каждое слово из буфера передачи используется для одной передачи по одному каналу 1 – режим распаковки включен. Слово из буфера передачи передается двумя посылками (по левому и правому каналу). При этом разрядность передаваемых слов не должна превышать 16 бит	RW	0
24:20	TWORDLEN	Длина передаваемого слова: Число бит в передаваемом слове равно TWORDLEN + 1. TWORDLEN должно быть больше 0.	RW	5'b0
19	TMBF	Порядок передачи бит: 0 – младшим битом вперед 1 – старшим битом вперед	RW	1
18	TCSNEG	Полярность управляющего сигнала передатчика: При TDSPMODE=0: TCSNEG = 0 – Левый канал передаётся с высоким уровнем TWS TCSNEG = 1 – Левый канал передаётся с низким уровнем TWS каждый фронт контрольного сигнала является активным и инициирует передачу нового слова. При TDSPMODE=1: задаёт полярность активного фронта: TCSNEG = 0 –передний фронт активный; TCSNEG = 1 –задний фронт активный;	RW	0
17:12	TWORDCNT	Число слов во фрейме: Определяет число передаваемых в течении одного фрейма слов. Число передаваемых слов равно TWORDCNT + 1. Число бит, передаваемых в пределах одного фрейма, равно (TWORDCNT + 1)*(TWORDLEN+1) При TPACK=1 обязательно TWORDCNT=0	RW	0
11	TDEL	Задержка начала передачи данных на такт: 0 – выдача первого бита передаваемого слова начинается по первому после активного фронта управляющего сигнала TWS фронту выдачи такового сигнала TCLK (используется для передачи в форматах Left-Justified и Right-Justified) 1 – выдача первого бита передаваемого слова начинается по второму после активного фронта управляющего сигнала TWS фронту выдачи такового сигнала TCLK (используется для передачи в формате I2S)	RW	0

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
10	TNEG	Полярность тактового сигнала передатчика: Задает исходное состояние вывода TCLK и фронт, по которому осуществляется выдача данных передатчиком (фронт выдачи) 0 – выдача данных по переднему фронту TCLK. 1 – выдача данных по заднему фронту TCLK. Исходное состояние TCLK = TNEG.	RW	0
9	TDSPMODE	Формат передачи данных: 0 – передача в формате I2S 1 – передача в формате DSP	RW	0
8:4	-	Резерв	-	0
3	-	В режиме I2S не используется	-	0
2	TD_ZER_EN	Обнуление избыточных бит передаваемого слова: 0 – Если длина слова меньше размеров окна, отведенного под передачу слова, после передачи всех бит слова на внешней шине данных остаётся значение нулевого бита передаваемого слова. 1 – Если длина слова меньше размеров окна, отведенного под передачу слова, после передачи всех бит слова на внешнюю шину данных подаётся 0, вплоть до начала передачи следующего слова. ВНИМАНИЕ! Режим с включенным обнулением избыточных бит при передаче слова корректно функционирует только при условии, что частота последовательного порта TCLK <= CLK/4, где CLK – рабочая частота, подаваемая на порт, со стороны системы.	RW	0
1	TMODE	Режим работы передатчика: 0 – режим I2S 1 – режим SPI	RW	0
0	TEN	Разрешение работы передатчика: 0 – передатчик выключен 1 – передатчик включен	RW	0

37.4.6 Регистр состояния приёмника RSR (режим I2S)

Таблица 37.13. Назначение разрядов регистра RSR в режиме I2S

Номер разряда	Условное обозначение	назначение	Доступ	Исходное состояние
31:28	-	резерв	-	0
27:24	RB_DIFF	Количество принятых 64-разрядных слов в буфере приёма (max 8).	R	0
23:19	-	Резерв	-	0
18:16	RLEV	Порог прерывания от буфера приёма: Прерывание формируется если число принятых 64-х разрядных слов больше RLEV	RW	7
15:11	-	Резерв	-	0
10	RXBUF	Результирующее прерывание MFBSP_RXBUF	R	0
9	RXBUF_D	Прерывание MFBSP_RXBUF без механизма автоматического сброса при чтении RSR	R	0
8	RXBUF_R	Прерывание MFBSP_RXBUF с механизмом автоматического сброса при чтении RSR	R	0
7	RRUN	Идёт приём: 0 – приёмник в состоянии ожидания 1 – идёт приём очередного слова	R	0

Номер разряда	Условное обозначение	назначение	Доступ	Исходное состояние
6	RERR	Ошибка передачи: 0 – приём проходил в штатном режиме 1 - была запись в полный буфер приёма (потеря данных). Флаг сбрасывается записью 0 в 6-й разряд регистра RSR.	RW	0
5	RSBF	Буфер пересинхронизации в направлении приёма полон: 0 – буфер пересинхронизации в направлении приёма не полон 1 – буфер пересинхронизации в направлении приёма полон	R	0
4	RSBE	Буфер пересинхронизации в направлении приёма пуст: 0 – буфер пересинхронизации в направлении приёма не пуст 1 – буфер пересинхронизации в направлении приёма пуст	R	1
3	RBHL	Достигнут порог прерывания в буфере приёма: 1 – число 64-х разрядных слов в буфере приёма больше чем задано в RLEV 0 – число 64-х разрядных слов в буфере приёма меньше либо равно RLEV	R	0
2	RBHF	Буфер приёма полон на половину или более: 1 – буфер приёма заполнен на половину или больше (из буфера приёма можно считать как минимум 4 слова) 0 – буфер приёма заполнен меньше, чем наполовину	R	0
1	RBF	Буфер приёма полон: 0 – буфер приёма не полон 1 – буфер приёма полон	R	0
0	RBE	Буфер приёма пуст: 0 – буфер приёма не пуст 1 – буфер приёма пуст	R	1

37.4.7 Регистр состояния передатчика TSR (режим I2S)

Таблица 37.14. Назначение разрядов регистра TSR в режиме I2S

Номер разряда	Условное обозначение	назначение	Доступ	Исходное состояние
31:28	-	резерв	-	0
27:24	TB_DIFF	Количество свободных 64-разрядных позиций в буфере передачи (в буфер передачи можно записать еще TB_DIFF 64-разрядных слов).	R	8
23	-	Резерв	-	0
22:20	TBES	Эффективный размер буфера передачи Актуален только для режима работы с DMA. Значение TBES+1 – определяет максимальный объем буфера передачи. Т.е. в режиме работы с DMA буфер передачи не может быть заполнен больше, чем на TBES+1 64 разрядных слов.	RW	7
19	-	Резерв	-	0
18:16	TLEV	Порог прерывания от буфера передачи: Прерывание формируется если число 64-х разрядных слов в буфере передачи меньше либо равно TLEV. В режиме передачи данных с использованием DMA определяет степень заполнения буфера передачи, при которой происходит запись в буфер очередной пачки данных	RW	0
15:11	-	Резерв	-	0

Номер разряда	Условное обозначение	назначение	Доступ	Исходное состояние
10	TXBUF	Результирующее прерывание MFBSP_TXBUF	R	0
9	TXBUF_D	Прерывание MFBSP_TXBUF без механизма автоматического сброса при чтении TSR	R	0
8	TXBUF_R	Прерывание MFBSP_TXBUF с механизмом автоматического сброса при чтении TSR	R	0
7	TRUN	Идёт передача: 0 – передатчик в состоянии ожидания 1 – идёт передача очередного слова	R	0
6	TERR	Ошибка передачи: 0 – передача проходила в штатном режиме 1 - было чтение из пустого буфера передачи (передача некорректных данных). Флаг сбрасывается записью 0 в 6-й разряд регистра TSR.	RW	0
5	TSBF	Буфер пересинхронизации в направлении передачи полон: 0 – буфер пересинхронизации в направлении передачи не полон 1 – буфер пересинхронизации в направлении передачи полон	R	0
4	TSBE	Буфер пересинхронизации в направлении передачи пуст: 0 – буфер пересинхронизации в направлении передачи не пуст 1 – буфер пересинхронизации в направлении передачи пуст	R	1
3	TBLL	Достигнут порог прерывания в буфере передачи: 1 – число 64-х разрядных слов в буфере передачи меньше либо равно TLEV 0 – число 64-х разрядных слов в буфере передачи больше TLEV	R	1
2	TBHF	Буфер передачи заполнен на половину или более: 1 – буфер передачи заполнен на половину или больше 0 – буфер передачи заполнен меньше чем на половину (в буфер передачи можно записать еще как минимум 4 слова)	R	0
1	TBF	Буфер передачи полон: 0 – буфер передачи не полон 1 – буфер передачи полон	R	0
0	TBE	Буфер передачи пуст: 0 – буфер передачи не пуст 1 – буфер передачи пуст	R	1

37.4.8 Регистр управления темпом приёма RCTR_RATE (режим I2S)

Таблица 37.15. Назначение разрядов регистра RCTR_RATE в режиме I2S

Номер разряда	Условное обозначение	назначение	Доступ	Исходное состояние
31:16	RCS_RATE	Делитель частоты управляющего сигнала приёмника: Задаёт частоту управляющего сигнала приёмника, определяемую, как RCLK/((RCS_RATE+1)*2), где RCLK – частота тактового сигнала приёмника RCS_RATE обязательно должно быть больше либо равно (RWORDLEN+1)*(RWORDCNT+1)-1	RW	0
15:12	-	В режиме I2S не используется	-	0
11:10	-	Резерв	-	0

9:0	RCLK_RATE	Делитель частоты приёмника: В случае, если частота формируется самим приёмником, определяет частоту приёмника RCLK = CLK/((RCLK_RATE+1)*2), где CLK – частота, подаваемая на порт со стороны системы.	RW	0

37.4.9 Регистр управления темпом передачи TCTR_RATE (режим I2S)

Таблица 37.16. Назначение разрядов регистра TCTR_RATE в режиме I2S

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:16	TCS_RATE	Делитель частоты управляющего сигнала передатчика: Задаёт частоту управляющего сигнала передатчика, определяемую как TCLK/((RCS_RATE+1)*2), где TCLK – частота тактового сигнала передатчика. TCS_RATE обязательно должно быть больше либо равно (TWORDLEN+1)*(TWORDCNT+1)-1	RW	0
15:12	-	В режиме I2S не используется	-	0
11:10	-	Резерв	-	0
9:0	TCLK_RATE	Делитель частоты передатчика: В случае, если частота формируется самим передатчиком, определяет частоту передатчика TCLK = CLK/((TCLK_RATE+1)*2), где CLK – частота, подаваемая на порт со стороны системы.	RW	0

37.4.10 Псевдорегистр TSTART (режим I2S)

Таблица 37.17. Назначение разрядов регистра TSTART в режиме I2S

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:1	-	Резерв	-	0
0	TEN	Разрешение работы передатчика: 0 – передатчик выключен 1 – передатчик включен Доступ к полю TEN регистра TCTR без изменения настроек TCTR	RW	0

37.4.11 Псевдорегистр RSTART (режим I2S)

Таблица 37.18. Назначение разрядов регистра RSTART в режиме I2S

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:1	-	Резерв	-	0
0	REN	Разрешение работы передатчика: 0 – приемник выключен 1 – приемник включен Доступ к полю REN регистра RCTR без изменения настроек RCTR	RW	0

37.4.12 Регистр аварийного управления портом EMERG_MFBSP (режим I2S)

Таблица 37.19. Назначение разрядов регистра EMERG_MFBSP в режиме I2S

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:6	-	Резерв	-	0
5	RX_DBG	Включение аварийной прокачки данных канала DMA направления приема. 0 – работа в обычном режиме 1 – сигнал готовности передачи данных по DMA каналу направления приема постоянно удерживается в 1 (по DMA принимаются невалидные данные)	RW	0
4	TX_DBG	Включение аварийной прокачки данных канала DMA направления передачи. 0 – работа в обычном режиме 1 – сигнал готовности передачи данных по DMA каналу направления передачи постоянно удерживается в 1 (данные передаваемые DMA теряются)	RW	0
3	-	Резерв	-	0
2	RST_RXBUF	Сброс буфера приема последовательного порта и буфера пересинхронизации направления приема. Запись единицы в данный разряд приводит к сбросу буфера, после чего данный бит снова устанавливается в 0.	RW	0
1	RST_TXBUF	Сброс буфера передачи последовательного порта и буфера пересинхронизации направления передачи. Запись единицы в данный разряд приводит к сбросу буфера, после чего данный бит снова устанавливается в 0.	RW	0
0	RST_LPTBUF	Сброс буфера линкового порта и буфера пересинхронизации направления приёма. Запись единицы в данный разряд приводит к сбросу буфера, после чего данный бит снова устанавливается в 0.	RW	0

37.4.13 Регистр маски прерываний от порта IMASK (режим I2S)

Таблица 37.20. Назначение разрядов регистра IMASK в режиме I2S

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:15	-	Резерв	-	0
14	RXBUF_R_EN	Разрешение автоматического сброса прерывания MFBSP_RXBUF 0 – прерывание не сбрасывается при чтении RSR 1 – прерывание сбрасывается при чтении RSR	RW	1
13	RX_LEV_IRQ_EN	Разрешение прерывания по уровню заполнения буфера приема 0 – прерывание MFBSP_RXBUF не будет устанавливаться при превышении порога RLEV 1 - прерывание MFBSP_RXBUF будет устанавливаться при превышении порога RLEV	RW	1
12	RX_ERR_IRQ_EN	Разрешение прерывания при переполнении буфера приема 0 – прерывание MFBSP_RXBUF не будет устанавливаться при переполнении буфера приема	RW	1

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
		1 - прерывание MFBSP_RXBUF будет устанавливаться при переполнении буфера приема		
11:7	-	Резерв	-	0
6	TXBUF_R_EN	Разрешение автоматического сброса прерывания MFBSP_TXBUF 0 – прерывание не сбрасывается при чтении TSR 1 – прерывание сбрасывается при чтении TSR	RW	1
5	TX_LEV_IRQ_EN	Разрешение прерывания по уровню заполнения буфера передачи 0 – прерывание MFBSP_TXBUF не будет устанавливаться, если число слов в буфере передачи меньше порога TLEV 1 - прерывание MFBSP_TXBUF будет устанавливаться, если число слов в буфере передачи меньше порога TLEV	RW	1
4	TX_ERR_IRQ_EN	Разрешение прерывания при переполнении буфера приема 0 – прерывание MFBSP_TXBUF не будет устанавливаться при чтении из пустого буфера передачи 1 - прерывание MFBSP_TXBUF будет устанавливаться при чтении из пустого буфера передачи	RW	1
3:1	-	Резерв	-	0
0	LPT_IRQ_EN	Разрешение прерывания по запросу на обслуживание 0 – SRQ запрещено 1 – SRQ разрешено	RW	1

37.4.14 Структурная схема MFBSP для режима I2S

На Рисунок 37.9 представлена структурная схема MFBSP для режима I2S.

Включение режима I2S производится установкой бит LEN=0, SPI_I2S_EN=1, регистра CSR_MFBSP и TMODE = 0 регистра TCTR для передатчика, RMODE = 0 регистра RCTR для приёмника.

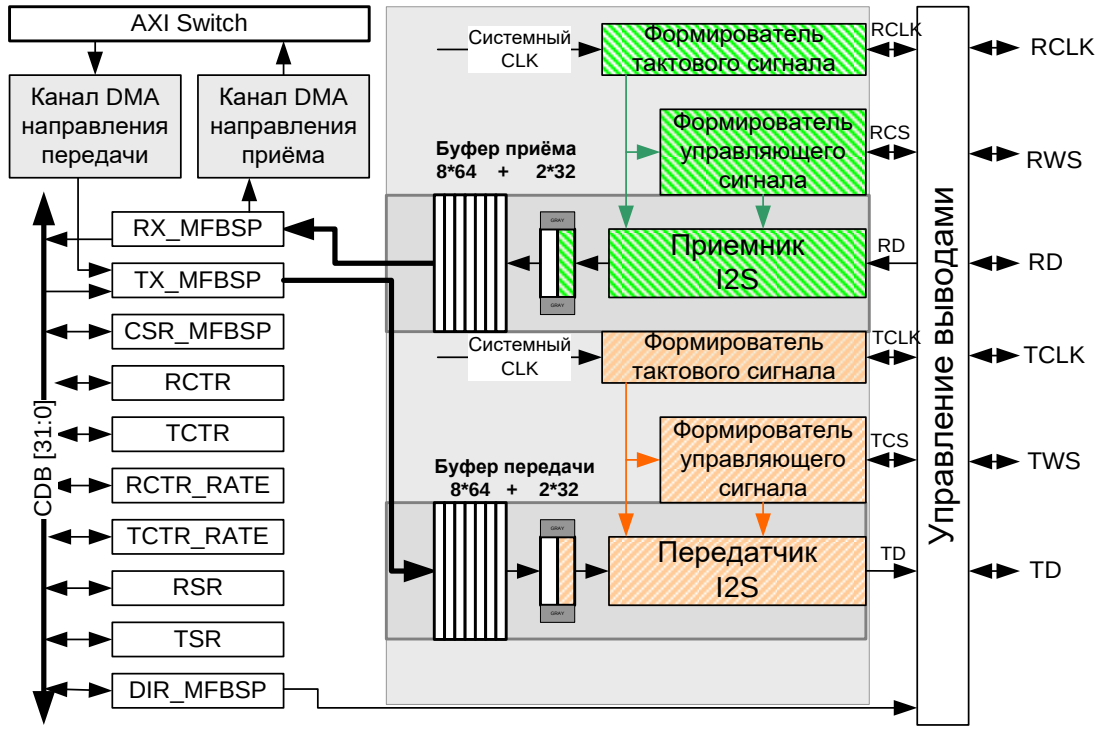


Рисунок 37.9. Структурная схема MFBSP для режима I2S

37.4.15 Варианты соединения порта с внешними устройствами

Программно управляя направлением выводов последовательного порта (см. описание регистра DIR_MFBSP) можно организовать множество вариантов соединения схемы с внешними устройствами через MFBSP (Рисунок 37.10,Рисунок 37.11,Рисунок 37.12).

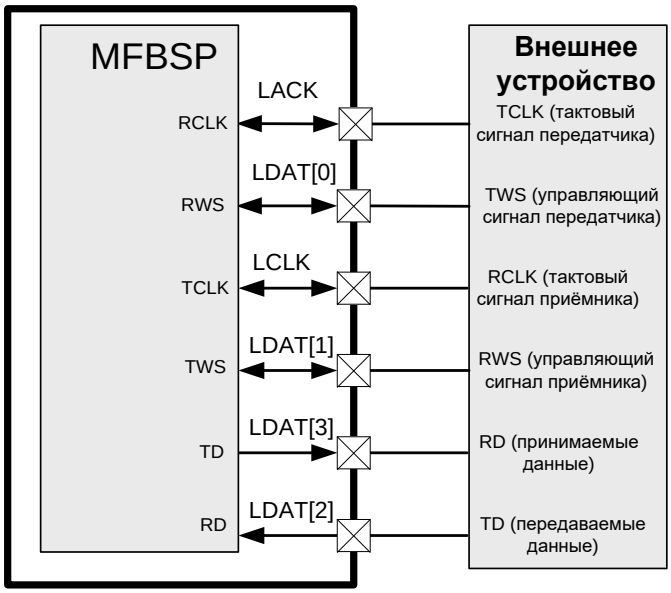


Рисунок 37.10. Соединение двух устройств по интерфейсу I2S в дуплексном режиме. Приёмник и передатчик независимые (задействовано 6 внешних выводов).

Направление выводов TCLK, TWS, RCLK и RWS может быть произвольным в зависимости от требований внешнего устройства (режим №3 по Таблица 37.4)

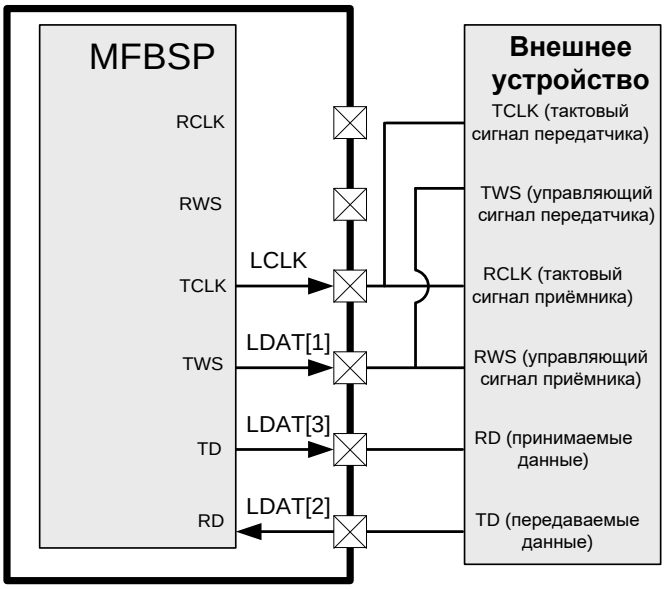


Рисунок 37.11. Соединение двух устройств по интерфейсу I2S в дуплексном режиме. Приёмник в зависимом от передатчика режиме (задействовано 4 внешних вывода) (режим №3 по Таблица 37.4)

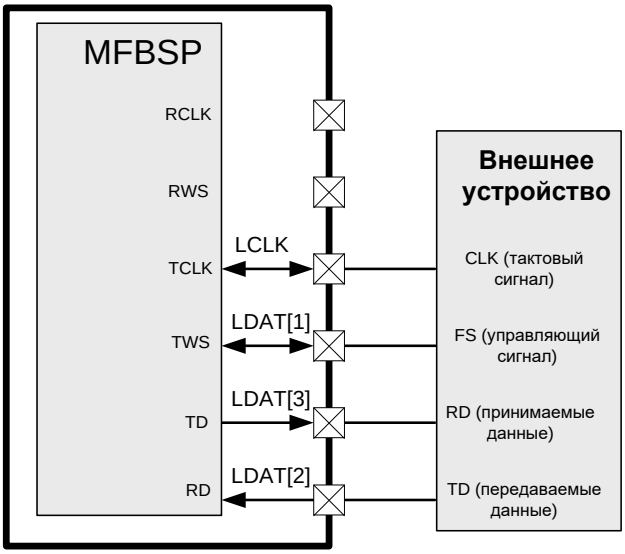


Рисунок 37.12. Соединение двух устройств по интерфейсу I2S в дуплексном режиме. Приёмник в зависимом от передатчика режиме (задействовано 4 внешних вывода). Как приёмником, так и передатчиком используются тактовый и управляющий сигналы с выводов TCLK и TWS. Направление выводов TCLK и TWS может быть произвольным в зависимости от требований внешнего устройства (режим №3 по Таблица 37.4)

37.4.16 Передача данных в режиме I2S

В режиме I2S возможна передача аудио данных с использованием сигнала выбора канала (бит (T/R)DSPMODE = 0). При этом программно задаётся полярность тактового сигнала,

полярность управляющего сигнала и наличие задержки выдачи данных относительно фронта управляющего сигнала (см. описание регистров TCTR и RCTR). На Рисунок 37.13 представлены временные диаграммы для данного режима.

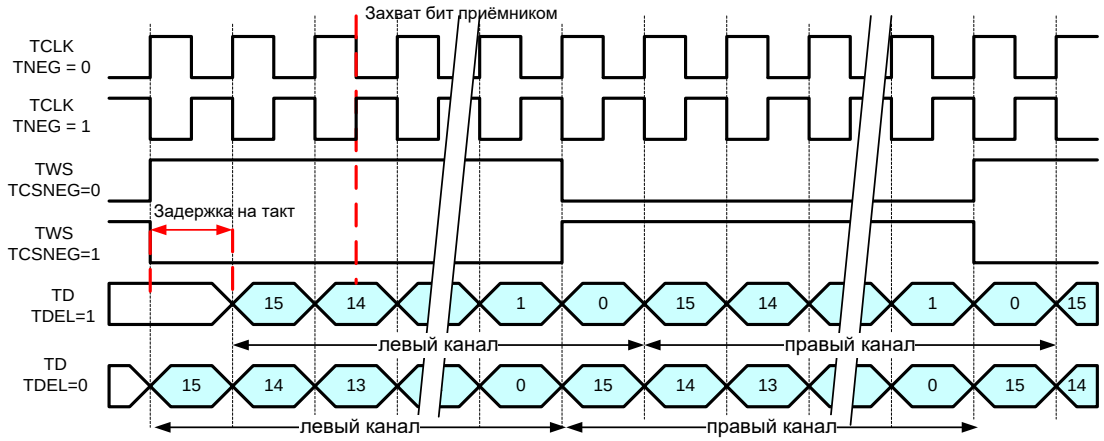


Рисунок 37.13. Передача в режиме I2S (формат I2S) TMODE = 0, TDSPMODE=0, TMBF = 1, TCS_RATE = TWORDLEN = 15 диаграммы тактового сигнала TCLK представлены для различных значений TNEG, диаграммы управляющего сигнала TWS представлены для различных значений TCSNEG, диаграммы для последовательных данных представлены для различных значений TDEL

В режиме I2S (бит (T/R)MODE = 0) также возможна передача последовательных слов с использованием сигнала синхронизации фрейма (бит (T/R)DSPMODE = 1). При этом программно задаётся полярность тактового сигнала, полярность активного фронта управляющего сигнала и наличие задержки выдачи данных относительно фронта управляющего сигнала (Рисунок 37.14).

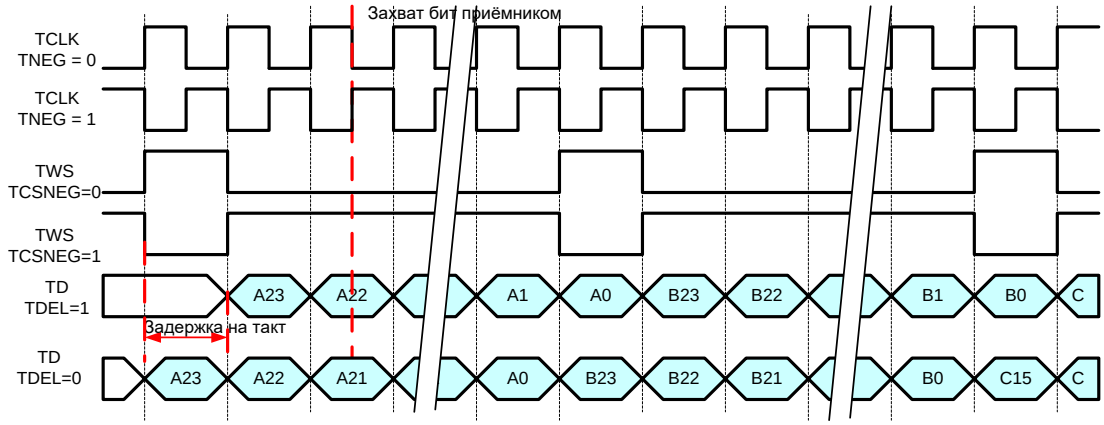


Рисунок 37.14. Передача в режиме I2S (формат DSP) TMODE = 0, TDSPMODE=1, TMBF = 1, TCS_RATE = TWORDLEN = 23 диаграммы тактового сигнала TCLK представлены для различных значений TNEG, диаграммы управляющего сигнала TWS представлены для различных значений TCSNEG, диаграммы для последовательных данных представлены для различных значений TDEL

Если управляющий сигнал формируется логикой MFBSP (вывод (T/R)WS – сконфигурирован как выход), то частота управляющего сигнала (либо частота импульсов

синхронизации в формате DSP) может задаваться программно от ICLK/2 до ICLK/(2*2^{16}), где ICLK – рабочая частота интерфейса TCLK для передатчика и RCLK для приемника (см. описание регистров TCTR_RATE и RCTR_RATE). Временные диаграммы для данного случая представлены на Рисунок 37.15.

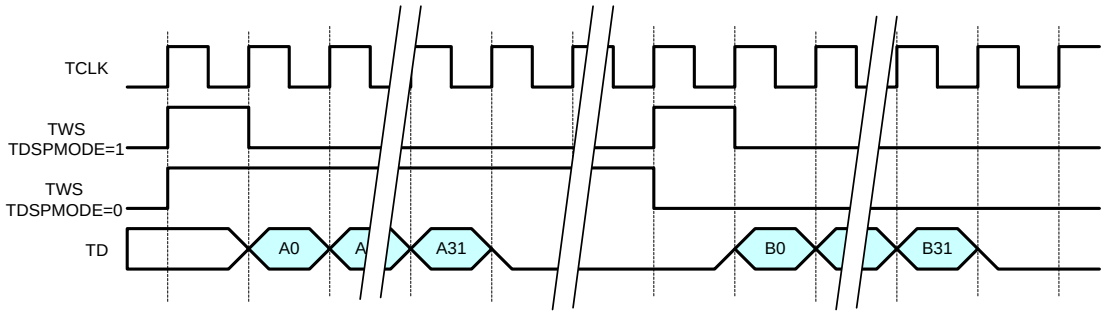


Рисунок 37.15. Передача в режиме I2S TMODE = 0, TMBF = 0, TWORDLEN = 31, TCS_RATE>TWORDLEN, TNEG = 0, TCSNEG=0, TDEL = 1. Диаграммы управляющего сигнала TWS представлены для различных значений TDSPMODE

MFBSP позволяет передавать от 1 до 64 слов в пределах одного фрейма (Рисунок 37.17). В этом случае с приходом сигнала синхронизации фрейма начинается передача первого слова, с передачей последнего бита первого слова из буфера передачи сразу считывается следующее слово и в следующем такте начинают передаваться биты очередного слова и так до тех пор, пока не будет передано число слов равное TWORDCNT+1. По окончании передачи последнего слова фрейма, порт ожидает очередного сигнала синхронизации фрейма. Сигнал синхронизации пришедший в момент, когда передача слов фрейма еще не закончилась игнорируется. Буфер передачи может вместить максимум 18 32-х разрядных слов, если в пределах фрейма передаётся больше 18 слов необходимо следить за тем, что бы буфер передачи был не пуст (при включенном канале DMA это происходит автоматически). Приёмник MFBSP аналогичным образом может принимать от 1 до 64-х слов в пределах одного фрейма.

В режиме I2S, при (T/R)MODE = 0, (T/R)DSPMODE =0) выполняется автоматическая синхронизация принимаемых и передаваемых данных таким образом, что первое слово переданное передатчиком будет передано в левый канал, а первое слово принятое приемником будет принято из левого канала (Рисунок 37.16).

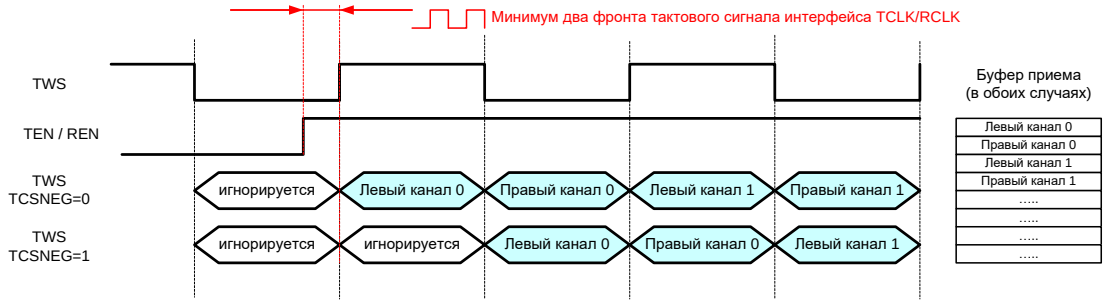


Рисунок 37.16. Синхронизация передаваемых и принимаемых данных по каналам (левый/правый) в режиме I2S после включения приемника или передатчика для различных значений TCSNEG

При работе порта в режиме I2S ((T/R)MODE = 0), в случае если используется управляющий сигнал, формируемый внешним устройством (порт в режиме ведомого), то, как для передатчика, так и для приемника после первого включения порта (TEN/REN=1), перед первым фронтом сигнала выбора слова / фрейм селекта (TWS) необходима подача как минимум двух импульсов тактового сигнала, необходимых для целей синхронизации. В противном случае первый импульс управляющего сигнала может быть проигнорирован (передача начнется со следующего активного фронта управляющего сигнала).

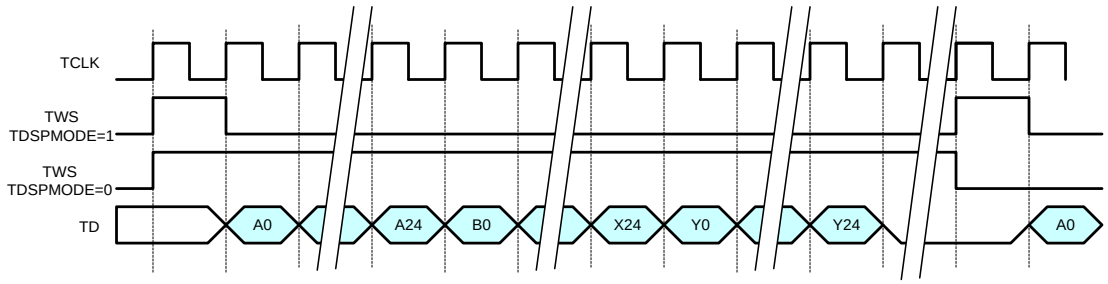


Рисунок 37.17. Передача в режиме I2S TMODE = 0, TMBF = 0, TWORDLEN = 24, TWORDCNT=Y-1, TCS_RATE+1>(TWORDLEN+1)*(TWORDCNT+1), TNEG = 0, TCSNEG=0, TDEL = 1. Диаграммы управляющего сигнала TWS представлены для различных значений TDSPMODE

В режиме I2S (только в формате I2S (T/R)DSPMODE=0) предусмотрен режим паковщика / распаковщика. В этом режиме 32 разрядные слова из буфера передачи автоматически разбиваются на 2 16-ти разрядных слова и передаются по разным каналам. Соответственно для приёмника два принятых по разным каналам слова группируются в одно 32-х разрядное слово, которое записывается в буфер приёма. В данном режиме длина передаваемого или принимаемого слова может быть в пределах от 2 до 16 бит. Порядок выдачи разбитого слова и порядок сборки определяется битами TCSNEG, TSWAP, RCSNEG, RSWAP. Данный режим возможен только при передаче одного слова во фрейме (TWORDCNT=RWORDCNT=0).

Пример настроек для передачи по интерфейсу FSB (CMX981): TMODE = RMODE = 0, TDSPMODE = RDSPMODE = 1, TMBF = RMBF = 1, TNEG = RNEG= 1, TDEL = RDEL = 1, TCSNEG = RCSNEG = 0, TCS_RATE >= TWORDLEN, RCS_RATE >= RWORDLEN. Приемник при этом должен быть независим от передатчика, т.е. RCS_CP = 0. Если шиной используется один тактовый сигнал для приема и передачи необходимо установить RCLK_CP в 1, в этом случае приемником будет использоваться тактовый сигнал передатчика.

37.4.17 Формирование тактовых сигналов приёмника (RCLK) и передатчика (TCLK)

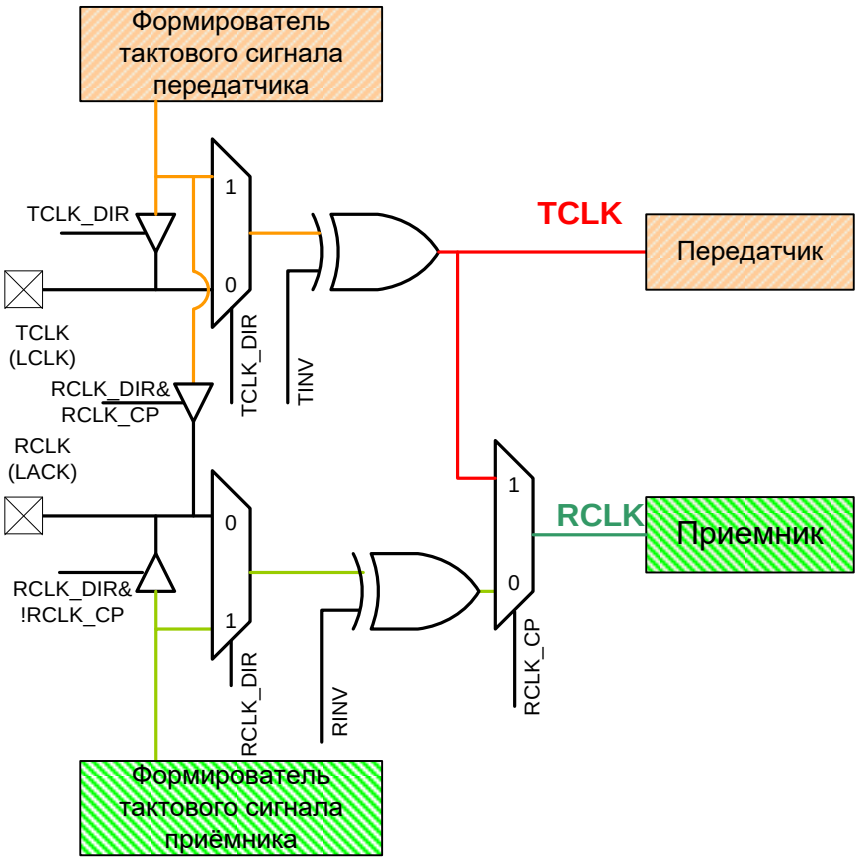


Рисунок 37.18. Схема формирования тактовых сигналов приёмника и передатчика в режиме I2S

На Рисунок 37.18 представлена схема формирования тактовых сигналов приёмника и передатчика в режиме I2S.

В зависимости от значения бита TCLK_DIR, тактовый сигнал передатчика TCLK может как формироваться самим передатчиком, так приниматься с внешнего вывода. В зависимости от значений бит TMODE, TNEG и TDEL тактовый сигнал либо передаётся передатчику без изменений, либо инвертируется.

В зависимости от значения бита RCLK_DIR, тактовый сигнал приёмника RCLK может как формироваться самим приёмником, так приниматься с внешнего вывода. В зависимости от значений бит RMODE, RNEG и RDEL тактовый сигнал либо передаётся приёмнику без изменений, либо инвертируется.

Если бит RCLK_CP установлен в 1, то тактовый сигнал приёмника копирует тактовый сигнал передатчика. Для корректной работы устройства в этом случае настройки полярности тактового сигнала приёмника и передатчика должны совпадать (TNEG=RNEG, TDEL=RDEL).

При RCLK_CP = 1 тактовый сигнал передатчика передаётся на внешний вывод приёмника, только если передатчик сам формирует тактовый сигнал и вывод тактового сигнала приёмника сконфигурирован как выход (TCLK_DIR=1, RCLK_DIR=1).

Если биты RCLK_CONT=1 и RCLK_DIR=1 то RCLK формируется непрерывно, пока установлен бит REN. Если RCLK_CONT=0 и RCLK_DIR=1 то RCLK формируется только до момента заполнения буфера приёма. Если RCLK_DIR=0, то RCLK принимается с внешнего вывода схемы.

Если биты TCLK_CONT=1 и TCLK_DIR=1 то TCLK формируется непрерывно, пока установлен бит TEN. Если TCLK_CONT=0 и TCLK_DIR=1 то TCLK формируется только в процессе передачи очередного слова. Если TCLK_DIR=0, то TCLK принимается с внешнего вывода схемы.

37.4.18 Формирование управляющих сигналов приёмника и передатчика в режиме I2S

На Рисунок 37.19 представлена схема формирования управляющих сигналов в режиме I2S.

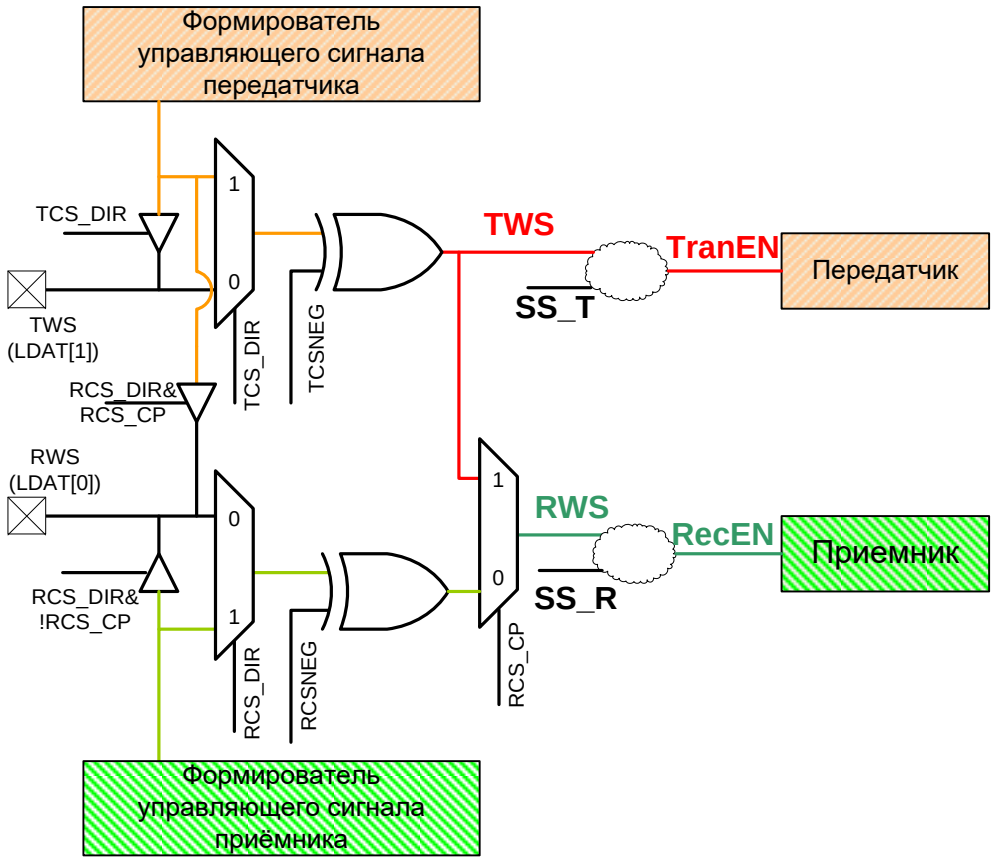


Рисунок 37.19. Схема формирования управляющих сигналов в режиме I2S

В зависимости от значения бита TCS_DIR, задающего направление вывода TWS, управляющий сигнал передатчика TWS может как формироваться самим передатчиком, так

приниматься с внешнего вывода. В зависимости от значения бита TCSNEG управляющий сигнал либо передаётся передатчику без изменений, либо инвертируется.

В зависимости от значения бита RCS_DIR, задающего направление вывода RWS, управляющий сигнал приёмника RCLK может как формироваться самим приёмником, так приниматься с внешнего вывода. В зависимости от значения бита RCSNEG управляющий сигнал либо передаётся приёмнику без изменений, либо инвертируется.

Если бит RCS_CP установлен в 1, то управляющий сигнал приёмника копирует управляющий сигнал передатчика. Для корректной работы устройства в этом случае настройки полярности управляющего сигнала приёмника и передатчика должны совпадать (TCSNEG=RCSNEG).

При RCS_CP = 1 управляющий сигнал передатчика передаётся на внешний вывод приёмника, только если передатчик сам формирует управляющий сигнал и вывод управляющего сигнала приёмника сконфигурирован как выход (TCS_DIR=1, RCS_DIR=1).

Если направление вывода RWS задано как выход и RCS_CONT=0, то управляющий сигнал RWS формируется до тех пор, пока не заполнится буфер приёма, если RCS_CONT=1 то, RWS формируется непрерывно, пока установлен бит REN. Если направление вывода задано как вход, управляющий сигнал RWS принимается от внешнего устройства. Если установлен бит RCS_CP, RWS копирует TWS, независимо от направления вывода.

Если направление вывода TWS задано как выход и TCS_CONT=0, то управляющий сигнал TWS формируется только во время передачи очередного слова, если TCS_CONT=1 TWS формируется непрерывно, пока установлен бит TEN. Если направление вывода задано как вход, управляющий сигнал TWS принимается от внешнего устройства.

37.4.19 Тракт передачи данных

На Рисунок 37.20 представлен тракт передачи данных для режима I2S.

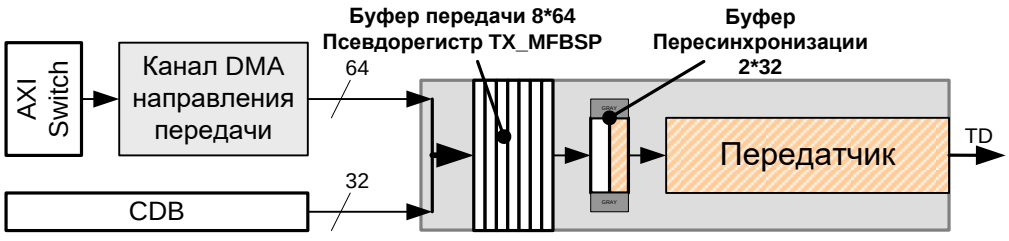


Рисунок 37.20. Тракт передачи данных для режима I2S

Что бы инициировать передачу данных по последовательному порту необходимо включить последовательный порт (SPI_I2S_EN=1) и передатчик (TEN=1), после чего либо начать производить запись передаваемых 32-х разрядных слов в буфер передачи по адресу псевдорегистра TX_MFBSP, либо включить канал DMA в направления передачи для

соответствующего порта (в этом случае обмен данными с портом будет вестись 64-х разрядными словами).

Данные записанные в буфер передачи автоматически перемещаются в буфер пересинхронизации направления передачи, если он не полон. Запись в буфер пересинхронизации направления передачи осуществляется на системной частоте CLK, чтение из буфера пересинхронизации осуществляется на частоте передатчика TCLK. Как только в буфере пересинхронизации оказалось хотя бы одно слово, передатчиком инициируется передача. Передатчиком производится последовательная выдача бит очередного 32-х разрядного слова до тех пор, пока число переданных бит не достигнет TWORDLEN+1, после чего производится считывание очередного слова из буфера пересинхронизации. По мере передачи слов в освобождающийся буфер пересинхронизации перемещается слово из буфера передачи. После выборки последнего слова из буфера передачи (буфер передачи пуст) в буфере пересинхронизации остаётся еще два слова. Фактическое окончание передачи можно идентифицировать по состоянию буфера пересинхронизации, либо считав бит TRUN регистра TSR.

Если управляющий сигнал формируется передатчиком, то при считывании последнего слова из буфера пересинхронизации передача останавливается. Передача продолжится только после того как в буфер пересинхронизации снова начнут поступать данные.

Если передатчик использует внешнюю частоту и внешний управляющий сигнал, в целях экономии мощности системная частота может быть установлена меньшей, чем внешняя частота передатчика, однако ее должно быть достаточно для того, чтобы успеть переместить очередное слово в буфер пересинхронизации (за время передачи одного слова должно быть хотя бы три импульса системной частоты CLK). Если внешний управляющий сигнал инициировал передачу слова при пустом буфере пересинхронизации устанавливается флаг ошибки передачи (TERR), в этом случае передаётся ошибочное слово. Если управляющий сигнал формируется самим передатчиком, системная частота может быть много меньше частоты передатчика, однако это скажется на скорости передачи данных.

Установка бита TERR в процессе передачи говорит о том, что порт произвел попытку чтения из пустого буфера передачи. Это значит, что передатчиком было передано некорректное слово.

В направлении передачи порт обладает буферизацией на 18 32-х разрядных слов. В случае передачи данных посредством DMA запись блоков данных в буфер передачи происходит до тех пор, пока буфер готов принять очередной блок, размер которого определяется битами WN, регистра CSR соответствующего канала DMA.

Степень заполнения буфера можно программно регулировать, используя биты TBES. В этом случае значение выражения TBES+1 – задает эффективный размер буфера передачи. Передача очередной пачки происходит только в случае, если при записи этой пачки в буфер

передачи число 64 разрядных слов в буфере передачи не превысит TBES+1. При попытке передать пачку со значением WN > TBES, значение WN автоматически корректируется (большая пачка передается в несколько этапов пачками меньшего размера).

Так, при WN=0 и TBES=0 очередное 64-х разрядное слово будет подкачиваться в буфер передачи только если он пуст. В этом случае по окончании работы DMA в момент прерывания от соответствующего канала передатчику останется передать ещё 4 32-х разрядных слова (два 32-х разрядных слова в буфере пересинхронизации и одно 64-х разрядное слово в буфере передачи).

Установка бита SPI_I2S_EN в 0 приведет к программному сбросу передатчика, и все данные находящиеся в буфере передачи будут утеряны.

37.4.20 Тракт приёма данных

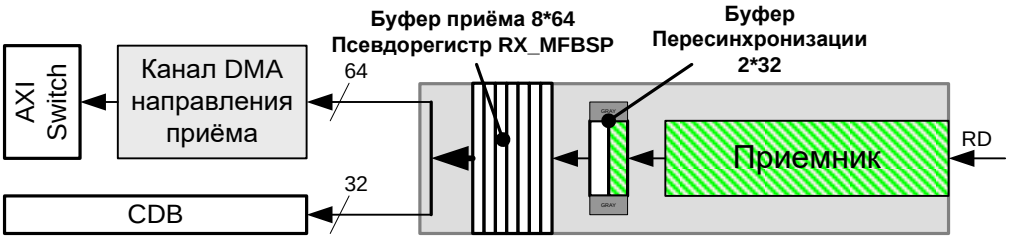


Рисунок 37.21. Тракт приёма данных в режиме I2S

На Рисунок 37.21 представлен тракт передачи данных для режима I2S.

Что бы перевести приёмник в режим готовности необходимо включить последовательный порт (SPI_I2S_EN=1) и приёмник (REN=1), после чего либо начать ожидание появления прочитанных данных в буфере приёма, либо включить канал DMA в направления приёма для соответствующего порта.

Приёмник принимает последовательные биты, поступающие с внешнего вывода до тех пор, пока число принятых бит не достигнет значения RWORDLEN+1. После этого принятое 32-х разрядное слово (если RWORDLEN<31 незадействованные биты обнуляются) перемещается в буфер пересинхронизации. Запись в буфер пересинхронизации направления приёма осуществляется на частоте приёмника RCLK, чтение из буфера пересинхронизации осуществляется на системной частоте CLK. Из буфера пересинхронизации принятое слово автоматически перемещается в буфер приёма, если он не полон. Если в буфере приёма есть хотя бы одно 32-х разрядное слово, то принятые 32-х разрядные слова можно считывать, обращаясь по адресу псевдорегистра RX_MFBSP. Принимать данные можно также включив соответствующий порту канал DMA направления приёма (в этом случае обмен данными с портом осуществляется 64-х разрядными словами).

Если приёмник использует внешнюю частоту, то в целях экономии мощности системная частота может быть установлена меньшей, чем внешняя частота приёмника, однако ее должно быть достаточно для того, чтобы успеть переместить очередное слово из буфера пересинхронизации (за время приёма одного слова должно быть хотя бы три импульса системной частоты CLK). Если при заполненном буфере пересинхронизации приёмником был произведен приём очередного слова и инициирована попытка записи в буфер пересинхронизации, устанавливается флаг ошибки приёма (RERR), а последнее принятое слово теряется.

Установка бита RERR в процессе передачи говорит о том, что порт произвел попытку записи в полный буфер приёма. Это значит, что принятое слово было потеряно.

В направлении приёма порт обладает буферизацией на 18 32-х разрядных слов. В случае приёма данных посредством DMA чтение блоков данных из буфера приёма происходит до тех пор, пока в буфере приёма достаточно слов для чтения очередного блока, размер которого определяется битами WN, регистра CSR соответствующего канала DMA. DMA обмены возможны только 64 разрядными словами, таким образом, если было принято нечетное количество 32-х разрядных слов, после окончания работы DMA необходимо прочитать оставшееся слово, обратившись к псевдорегистру RX_MFBSP.

Установка бита SPI_I2S_EN в 0 приведет к программному сбросу приёмника и все данные находящиеся в буфере приёма будут утеряны.

37.4.21 Прерывания от последовательного порта

Прерывание MFBSP_RXBUF устанавливается, в случае если включен приемник (I2S_SPI_EN=1, REN = 1) и в буфер приёма записано количество слов большее, чем установлено уровнем прерывания RLEV, либо произошла ошибка приема (RERR = 1).

Прерывание MFBSP_TXBUF устанавливается, в случае если включен передатчик (I2S_SPI_EN=1, REN = 1) и в буфере передачи осталось количество слов меньшее, либо равное чем установлено уровнем прерывания TLEV, либо произошла ошибка передачи (TERR = 1).

37.5 Работа MFBSP в режиме SPI

37.5.1 Назначение последовательного порта в режиме SPI

Режим SPI буферизированного последовательного порта предназначен для организации дуплексного обмена последовательными данными с внешними устройствами.

Порт в режиме SPI позволяет одновременно передавать и принимать последовательные данные. Приемник и передатчик контроллера могут настраиваются независимо, при этом возможен перевод приёмника в зависимое от передатчика состояние.

Поддерживается независимое задание направления каждого из выводов порта, осуществляемое установкой соответствующих бит регистра DIR_MFBSP. Однако для режима SPI имеется ограничение: направление выводов тактового сигнала и сигнала выбора ведомого должно совпадать.

В режиме ведущего устройства к MFBSP параллельно может быть подключено до двух ведомых SPI устройств.

Формирование сигнала выбора ведомого возможно как в автоматическом, так и в программном режиме. В автоматическом режиме после передачи каждой группы слов (число слов в группе может принимать значения от 1 до 64) сигнал выбора ведомого возвращается в высокое состояние. При программном управлении сигналами выбора ведомого данные сигналы изменяются посредством записи в контрольный регистр передатчика.

В данной реализации порта существует ограничение на выбор направления выводов в режиме SPI: тактовый и управляющий сигналы в режиме SPI должны быть либо оба заданы как вход, либо оба заданы как выход;

В данной реализации порта не предусмотрена возможность соединения нескольких микропроцессоров по цепочке с использованием SPI интерфейса. Микропроцессор может только управлять загрузкой последовательных данных в другие ведомые устройства, соединенные по цепочке.

В данной реализации порта в режиме ведомого устройства сигнал выбора ведомого предварительно пересинхронизируется на внутреннюю частоту порта, поэтому для устойчивой работы порта в режиме ведомого SPI устройства уровень сигнала SS, если необходима его установка в 1 между передачами, должен удерживаться как минимум два периода внутренней частоты CLK. Поэтому, если приемник работает в зависимом от передатчика режиме (RCS_CP=1, RCLK_CP=1), передатчик работает на максимальной частоте (TCLK_RATE=0) и формирует сигнал SS в автоматическом режиме (SS_DO=0, TCS_DIR=1), необходимо установить значение TSS_RATE>=1 чтобы удерживать сигнал SS в высоком уровне как минимум два периода внутренней частоты CLK.

37.5.2 Регистр управления и состояния CSR_MFBSP (режим SPI)

Регистр CSR_MFBSP (Таблица 37.21) используется для включения режима последовательного порта и разрешения прерываний от MFBSP.

Таблица 37.21. Назначение разрядов регистра CSR_MFBSP в режиме SPI

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:15	-	Резерв	-	0
14:11	-	В режиме SPI не используется	-	0
10	-	Резерв	-	0
9	SPI_I2S_EN	Включение режима SPI/I2S: 0 – Работа в режиме LPORT 1 – Работа в режиме SPI/I2S	RW	0
8:5	-	В режиме I2S не используется	-	0
4:3	LSTAT	Состояние буфера: При LTRAN = 0 показывает состояние буфера приёма При LTRAN = 1 показывает состояние буфера передачи 00 – буфер пуст; 10 – буфер не пуст; 11 – буфер полон.	R	0
2	-	В режиме I2S не используется	-	0
1	LTRAN	Назначение бит LSTAT: 0 - LSTAT отображает состояние буфера приёма 1 - LSTAT отображает состояние буфера передачи	RW	0
0	LEN	В режиме SPI должен быть установлен в 0	RW	0

37.5.3 Регистр управления направлением выводов DIR_MFBSP (режим SPI)

Регистр управления направлением выводов DIR_MFBSP (Таблица 37.22) предназначен для индивидуальной настройки направления каждого вывода последовательного порта.

Таблица 37.22. Назначение разрядов регистра DIR_MFBSP в режиме SPI

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
9:6	LDAT_DIR[7:4]	Направление выводов LDAT[7:4]	RW	0
5	TD_DIR	Направление вывода MOSI: 0 – MOSI – вход (при RD_DIR = 1 последовательные данные принимаются со входа MOSI - эквивалент SDI) 1 – MOSI - выход (MOSI – является выходом для передачи последовательных данных и является эквивалентом SDO)	RW	0
4	RD_DIR	Направление вывода MISO: 0 – MISO – вход (последовательные данные принимаются со входа MISO - эквивалент SDI) 1 – MISO - выход (MISO – является выходом для передачи последовательных данных и является эквивалентом SDO)	RW	0
3	TCS_DIR	Направление вывода SS[0]:	RW	0

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
		0 – SS[0] – вход (управляющий сигнал для передатчика снимается с вывода SS[0]) 1 – SS[0] - выход, управляющий сигнал формируется передатчиком		
2	RCS_DIR	Направление вывода SS[1]: 0 – SS[1] – вход (управляющий сигнал для приёмника снимается с вывода SS[1]) 1 – SS[1] - выход, в этом случае на SS[1] в зависимости от состояния бита RCS_CP подаются управляющие сигналы, формируемые либо приемником, либо передатчиком	RW	0
1	TCLK_DIR	Направление вывода TSCK: 0 – TSCK – вход (тактовый сигнал TSCK принимается от внешнего источника) 1 – TSCK – выход (тактовый сигнал TSCK формируется передатчиком)	RW	0
0	RCLK_DIR	Направление вывода RSCK: 0 – RSCK – вход (тактовый сигнал RSCK принимается от внешнего источника) 1 – RSCK – выход (тактовый сигнал RSCK формируется приёмником)	RW	0

Примечание. При RD_DIR = 0 и TD_DIR = 0 данные снимаются с MISO, при RD_DIR = 1 и TD_DIR = 1 на MOSI и MISO выдаются одинаковые данные с передатчика.

37.5.4 Регистр управления приёмником RCTR (режим SPI)

Таблица 37.23. Назначение разрядов регистра RCTR в режиме SPI

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:30	-	Резерв	-	0
29	-	В режиме SPI не используется	-	0
28	-	В режиме SPI не используется	-	0
27	-	В режиме SPI не используется	-	0
26	RSIGN	Значение заполнителя: Если длина принимаемого слова меньше 32 при отключенном паковщике или меньше 16 при включенном паковщике, то неиспользуемые биты принятого слова заполняются При RSIGN = 0 нулями При RSIGN = 1 значением старшего разряда в принятом слове	RW	0
25	RPACK	В режиме SPI обязательно RPACK=0.	RW	0
24:20	RWORDLEN	Длина принимаемого слова: Число бит в принимаемом слове равно RWORDLEN + 1. RWORDLEN должно быть больше 0.	RW	5’b0
19	RMBF	Порядок передачи бит: 0 – младшим битом вперед 1 – старшим битом вперед	RW RW	1
18	-	В режиме SPI не используется	-	0
17:12	RWORDCNT	Число слов во фрейме: Определяет число принимаемых в течении одного фрейма слов. Число принимаемых слов равно RWORDCNT + 1.	RW	0

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
		Число бит, принимаемых в пределах одного фрейма, равно (RWORDCNT + 1)*(RWORDLEN+1) Во время приёма фрейма состояние сигнала выбора ведомого не меняется.		
11	RDEL	Задержка начала приёма данных на пол такта: (Эквивалентно CPHA в спецификации Motorola). Задает фронт, по которому производится захват данных приёмником (фронт приёма). Ниже приведено соответствие полярности фронта приёма значениям бит RNEG, RDEL: RNEG = 0, RDEL = 0 – захват по переднему фронту RSCK RNEG = 0, RDEL = 1 – захват по заднему фронту RSCK RNEG = 1, RDEL = 0 – захват по заднему фронту RSCK RNEG = 1, RDEL = 1 – захват по переднему фронту RSCK	RW	0
10	RNEG	Полярность тактового сигнала приёмника: (эквивалентно CPOL в спецификации Motorola). Задает исходное состояние вывода RSCK и фронт, по которому производится захват данных приёмником (фронт приёма). Ниже приведено соответствие полярности фронта приёма значениям бит RNEG, RDEL: RNEG = 0, RDEL = 0 – захват по переднему фронту RSCK RNEG = 0, RDEL = 1 – захват по заднему фронту RSCK RNEG = 1, RDEL = 0 – захват по заднему фронту RSCK RNEG = 1, RDEL = 1 – захват по переднему фронту RSCK Исходное состояние RSCK = RNEG.	RW	0
9	-	В режиме SPI не используется	-	0
8:4	-	резерв	-	0
3	RCS_CP	Управление сигналом выбора ведомого приёмника: 0 – сигнал SS[1] принимается приёмником с внешнего вывода или формируется самим приёмником. 1 - сигнал SS[1] формируется передатчиком и является сигналом выбора ведомого устройства 1. Приёмник осуществляет приём данных синхронно с передатчиком. (в этом случае RCLK_CP должно быть так же в 1).	RW	0
2	RCLK_CP	Дублирование сигнала RSCK: 0 – RSCK формируется или принимается независимо от передатчика 1 – RSCK приёмника дублирует TSCK передатчика (в этом случае RCS_CP должно быть так же в 1).	RW	0
1	RMODE	Режим работы приёмника: 0 – режим I2S 1 – режим SPI	RW	0
0	REN	Разрешение работы приёмника: 0 – приемник выключен 1 – приемник включен	RW	0

37.5.5 Регистр управления передатчиком TCTR (режим SPI)

Таблица 37.24. Назначение разрядов регистра TCTR в режиме SPI

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31	SS[1]	биты управления шиной Slave Select:	RW	0

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
30	SS[0]	Позволяют выбрать одно из двух подключенных ведомых устройств. При SS_DO = 0 установка соответствующего бита SS в 1 означает выбор ведомого устройства, с которым будет производится обмен данными При SS_DO = 1 значения бит SS передаются на выводы SS напрямую	RW	0
29	-	В режиме SPI не используется	-	0
28	-	В режиме SPI не используется	-	0
27	-	В режиме SPI не используется	-	0
26	-	Резерв	-	0
25	TPACK	В режиме SPI обязательно TPACK=0.	RW	0
24:20	TWORDLEN	Длина передаваемого слова: Число бит в передаваемом слове равно TWORDLEN + 1. TWORDLEN должно быть больше 0.	RW	5'b0
19	TMBF	Порядок передачи бит: 0 – младшим битом вперед 1 – старшим битом вперед	RW	1
18	-	В режиме SPI не используется	-	0
17:12	TWORDCNT	Число слов во фрейме: Определяет число передаваемых в течении одного фрейма слов. Число передаваемых слов равно TWORDCNT + 1. Число бит, передаваемых в пределах одного фрейма, равно (TWORDCNT + 1)*(TWORDLEN+1) Во время передачи фрейма состояние сигнала выбора ведомого не меняется.	RW	0
11	TDEL	Задержка начала передачи данных на пол такта: (Эквивалентно CPHA в спецификации Motorola). Задает фронт, по которому производится выдача данных передатчиком (фронт выдачи). Ниже приведено соответствие полярности фронта выдачи значениям бит TNEG, TDEL: TNEG = 0, TDEL = 0 – выдача по заднему фронту TSCK TNEG = 0, TDEL = 1 – выдача по переднему фронту TSCK TNEG = 1, TDEL = 0 – выдача по переднему фронту TSCK TNEG = 1, TDEL = 1 – выдача по заднему фронту TSCK	RW	0
10	TNEG	Полярность тактового сигнала передатчика: (эквивалентно CPOL в спецификации Motorola). Задает исходное состояние вывода TSCK и фронт, по которому производится выдача данных передатчиком (фронт выдачи). Ниже приведено соответствие полярности фронта выдачи значениям бит TNEG, TDEL: TNEG = 0, TDEL = 0 – выдача по заднему фронту TSCK TNEG = 0, TDEL = 1 – выдача по переднему фронту TSCK TNEG = 1, TDEL = 0 – выдача по переднему фронту TSCK TNEG = 1, TDEL = 1 – выдача по заднему фронту TSCK Исходное состояние TSCK = TNEG.	RW	0
9	-	В режиме SPI не используется	-	0
8:4	-	резерв	-	0
3	SS_DO	управление выводами SS: 0 – управление выводами SS производится в автоматическом режиме. С началом передачи вывод SS, для которого соответствующий бит SS, регистра TCRT	RW	0

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
		установлен в 1, переводится в низкое состояние, с окончанием передачи вывод SS переводится в высокое состояние. Если соответствующий выводу бит SS установлен в 0, вывод SS всегда находится в высоком состоянии. 1 – значения бит SS напрямую передаются на внешние выводы. В этом случае необходимо программное управление шиной SS в процессе передачи		
2	-	В режиме SPI не используется	-	0
1	TMODE	Режим работы передатчика: 0 – режим I2S 1 – режим SPI	RW	0
0	TEN	Разрешение работы передатчика: 0 – приемник выключен 1 – приемник включен	RW	0

37.5.6 Регистр состояния приёмника RSR (режим SPI)

Таблица 37.25. Назначение разрядов регистра RSR в режиме SPI

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:28	-	резерв	-	0
27:24	RB_DIFF	Количество принятых 64-разрядных слов в буфере приёма (мах 8).	R	0
23:19	-	Резерв	-	0
18:16	RLEV	Порог прерывания от буфера приёма: Прерывание формируется если число принятых 64-х разрядных слов больше RLEV	RW	7
15:11	-	Резерв	-	0
10	RXBUF	Результирующее прерывание MFBSP_RXBUF	R	0
9	RXBUF_D	Прерывание MFBSP_RXBUF без механизма автоматического сброса при чтении RSR	R	0
8	RXBUF_R	Прерывание MFBSP_RXBUF с механизмом автоматического сброса при чтении RSR	R	0
7	RRUN	Идёт приём: 0 – приёмник в состоянии ожидания 1 – идёт приём очередного слова	R	0
6	RERR	Ошибка передачи: 0 – приём проходил в штатном режиме 1 - была запись в полный буфер приёма (потеря данных). Флаг сбрасывается записью 0 в 6-й разряд регистра RSR.	RW	0
5	RSBF	Буфер пересинхронизации в направлении приёма полон: 0 – буфер пересинхронизации в направлении приёма не полон 1 – буфер пересинхронизации в направлении приёма полон	R	0
4	RSBE	Буфер пересинхронизации в направлении приёма пуст: 0 – буфер пересинхронизации в направлении приёма не пуст 1 – буфер пересинхронизации в направлении приёма пуст	R	1
3	RBHL	Достигнут порог прерывания в буфере приёма: 1 – число 64-х разрядных слов в буфере приёма больше чем задано в RLEV	R	0

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
		0 – число 64-х разрядных слов в буфере приёма меньше либо равно RLEV		
2	RBHF	Буфер приёма полон на половину или более: 1 – буфер приёма заполнен на половину или больше (из буфера приёма можно считать как минимум 4 слова) 0 – буфер приёма заполнен меньше, чем наполовину	R	0
1	RBF	Буфер приёма полон: 0 – буфер приёма не полон 1 – буфер приёма полон	R	0
0	RBE	Буфер приёма пуст: 0 – буфер приёма не пуст 1 – буфер приёма пуст	R	1

37.5.7 Регистр состояния передатчика TSR (режим SPI)

Таблица 37.26. Назначение разрядов регистра TSR в режиме SPI

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:28	-	резерв	-	0
27:24	TB_DIFF	Количество свободных 64-разрядных позиций в буфере передачи (в буфер передачи можно записать еще TB_DIFF 64-разрядных слов).	R	8
23	-	Резерв	-	0
22:20	TBES	Эффективный размер буфера передачи Актуален только для режима работы с DMA. Значение TBES+1 – определяет максимальный объем буфера передачи. Т.е. в режиме работы с DMA буфер передачи не может быть заполнен больше, чем на TBES+1 64 разрядных слов.	RW	7
19	-	Резерв	-	0
18:16	TLEV	Порог прерывания от буфера передачи: Прерывание формируется если число 64-х разрядных слов в буфере передачи меньше либо равно TLEV. В режиме передачи данных с использованием DMA определяет степень заполнения буфера передачи, при которой происходит запись в буфер очередной пачки данных	RW	0
15:11	-	Резерв	-	0
10	TXBUF	Результирующее прерывание MFBSP_TXBUF	R	0
9	TXBUF_D	Прерывание MFBSP_TXBUF без механизма автоматического сброса при чтении TSR	R	0
8	TXBUF_R	Прерывание MFBSP_TXBUF с механизмом автоматического сброса при чтении TSR	R	0
7	TRUN	Идёт передача: 0 – передатчик в состоянии ожидания 1 – идёт передача очередного слова	R	0
6	TERR	Ошибка передачи: 0 – передача проходила в штатном режиме 1 - было чтение из пустого буфера передачи (передача некорректных данных). Флаг сбрасывается записью 0 в 6-й разряд регистра TSR.	RW	0
5	TSBF	Буфер пересинхронизации в направлении передачи полон: 0 – буфер пересинхронизации в направлении передачи не полон	R	0

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
		1 – буфер пересинхронизации в направлении передачи полон		
4	TSBE	Буфер пересинхронизации в направлении передачи пуст: 0 – буфер пересинхронизации в направлении передачи не пуст 1 – буфер пересинхронизации в направлении передачи пуст	R	1
3	TBLL	Достигнут порог прерывания в буфере передачи: 1 – число 64-х разрядных слов в буфере передачи меньше либо равно TLEV 0 – число 64-х разрядных слов в буфере передачи больше TLEV	R	1
2	TBHF	Буфер передачи заполнен на половину или более: 1 – буфер передачи заполнен на половину или больше 0 – буфер передачи заполнен меньше чем на половину (в буфер передачи можно записать еще как минимум 4 слова)	R	0
1	TBF	Буфер передачи полон: 0 – буфер передачи не полон 1 – буфер передачи полон	R	0
0	TBE	Буфер передачи пуст: 0 – буфер передачи не пуст 1 – буфер передачи пуст	R	1

37.5.8 Регистр управления темпом приёма RCTR_RATE (режим SPI)

Таблица 37.27. Назначение разрядов регистра RCTR_RATE в режиме SPI

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:16	-	В режиме SPI не используется	-	0
15:12	RSS_RATE	Если сигнал SS формируется приёмником, то задает время удержания сигнала SS в высоком уровне между передачами слов. Время удержания SS определяется как TRCLK/2*(RSS_RATE+1), где TRCLK период тактового сигнала RCLK	RW	0
11:10	-	Резерв	-	0
9:0	RCLK_RATE	Делитель частоты приёмника: В случае, если частота формируется самим приёмником, определяет частоту приёмника RSCK= CLK/((RCLK_RATE+1)*2), где CLK – частота, подаваемая на порт со стороны системы.	RW	0

37.5.9 Регистр управления темпом передачи TCTR_RATE (режим SPI)

Таблица 37.28. Назначение разрядов регистра TCTR_RATE в режиме SPI

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:16	-	В режиме SPI не используется	-	0
15:12	TSS_RATE	Если сигнал SS формируется передатчиком, то задает время удержания сигнала SS в высоком уровне между передачами слов. Время удержания SS определяется как TTCLK/2*(TSS_RATE+1), где TTCLK период тактового сигнала TCLK	RW	0
11:10	-	Резерв	-	0

9:0	TCLK_RATE	Делитель частоты передатчика: В случае, если частота формируется самим передатчиком, определяет частоту передатчика TSCK = CLK/((TCLK_RATE+1)*2), где CLK – частота, подаваемая на порт со стороны системы.	RW	0

37.5.10 Псевдорегистр TSTART (режим SPI)

Таблица 37.29. Назначение разрядов регистра TSTART в режиме SPI

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:1	-	Резерв	-	0
0	TEN	Разрешение работы передатчика: 0 – передатчик выключен 1 – передатчик включен Доступ к полю TEN регистра TCTR без изменения настроек TCTR	RW	0

37.5.11 Псевдорегистр RSTART (режим SPI)

Таблица 37.30. Назначение разрядов регистра RSTART в режиме SPI

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:1	-	Резерв	-	0
0	REN	Разрешение работы передатчика: 0 – приемник выключен 1 – приемник включен Доступ к полю REN регистра RCTR без изменения настроек RCTR	RW	0

37.5.12 Регистр аварийного управления портом EMERG_MFBSP (режим SPI)

Таблица 37.31. Назначение разрядов регистра EMERG_MFBSP в режиме SPI

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:6	-	Резерв	-	0
5	RX_DBG	Включение аварийной прокачки данных канала DMA направления приема. 0 – работа в обычном режиме 1 – сигнал готовности передачи данных по DMA каналу направления приема постоянно удерживается в 1 (по DMA принимаются невалидные данные)	RW	0
4	TX_DBG	Включение аварийной прокачки данных канала DMA направления передачи. 0 – работа в обычном режиме 1 – сигнал готовности передачи данных по DMA каналу направления передачи постоянно удерживается в 1 (данные передаваемые DMA теряются)	RW	0
3	-	Резерв	-	0
2	RST_RXBUF	Сброс буфера приема последовательного порта и буфера пересинхронизации направления приема.	RW	0

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
		Запись единицы в данный разряд приводит к сбросу буфера, после чего данный бит снова устанавливается в 0.		
1	RST_TXBUF	Сброс буфера передачи последовательного порта и буфера пересинхронизации направления передачи. Запись единицы в данный разряд приводит к сбросу буфера, после чего данный бит снова устанавливается в 0.	RW	0
0	RST_LPTBUF	Сброс буфера линкового порта и буфера пересинхронизации направления приёма. Запись единицы в данный разряд приводит к сбросу буфера, после чего данный бит снова устанавливается в 0.	RW	0

37.5.13 Регистр маски прерываний от порта IMASK (режим SPI)

Таблица 37.32. Назначение разрядов регистра IMASK в режиме SPI

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:15	-	Резерв	-	0
14	RXBUF_R_EN	Разрешение автоматического сброса прерывания MFBSP_RXBUF 0 – прерывание не сбрасывается при чтении RSR 1 – прерывание сбрасывается при чтении RSR	RW	1
13	RX_LEV_IRQ_EN	Разрешение прерывания по уровню заполнения буфера приема 0 – прерывание MFBSP_RXBUF не будет устанавливаться при превышении порога RLEV 1 - прерывание MFBSP_RXBUF будет устанавливаться при превышении порога RLEV	RW	1
12	RX_ERR_IRQ_EN	Разрешение прерывания при переполнении буфера приема 0 – прерывание MFBSP_RXBUF не будет устанавливаться при переполнении буфера приема 1 - прерывание MFBSP_RXBUF будет устанавливаться при переполнении буфера приема	RW	1
11:7	-	Резерв	-	0
6	TXBUF_R_EN	Разрешение автоматического сброса прерывания MFBSP_TXBUF 0 – прерывание не сбрасывается при чтении TSR 1 – прерывание сбрасывается при чтении TSR	RW	1
5	TX_LEV_IRQ_EN	Разрешение прерывания по уровню заполнения буфера передачи 0 – прерывание MFBSP_TXBUF не будет устанавливаться, если число слов в буфере передачи меньше порога TLEV 1 - прерывание MFBSP_TXBUF будет устанавливаться, если число слов в буфере передачи меньше порога TLEV	RW	1
4	TX_ERR_IRQ_EN	Разрешение прерывания при переполнении буфера приема 0 – прерывание MFBSP_TXBUF не будет устанавливаться при чтении из пустого буфера передачи 1 - прерывание MFBSP_TXBUF будет устанавливаться при чтении из пустого буфера передачи	RW	1
3:1	-	Резерв	-	0

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
0	LPT_IRQ_EN	Разрешение прерывания по запросу на обслуживание 0 – SRQ запрещено 1 – SRQ разрешено	RW	1

37.5.14 Структурная схема MFBSP для режима SPI

На Рисунок 37.22 представлена структурная схема MFBSP для режима SPI.

Включение режима SPI производится установкой бит LEN=0, SPI_I2S_EN=1, TMODE = 1 (для передатчика), RMODE = 1 (для приёмника).

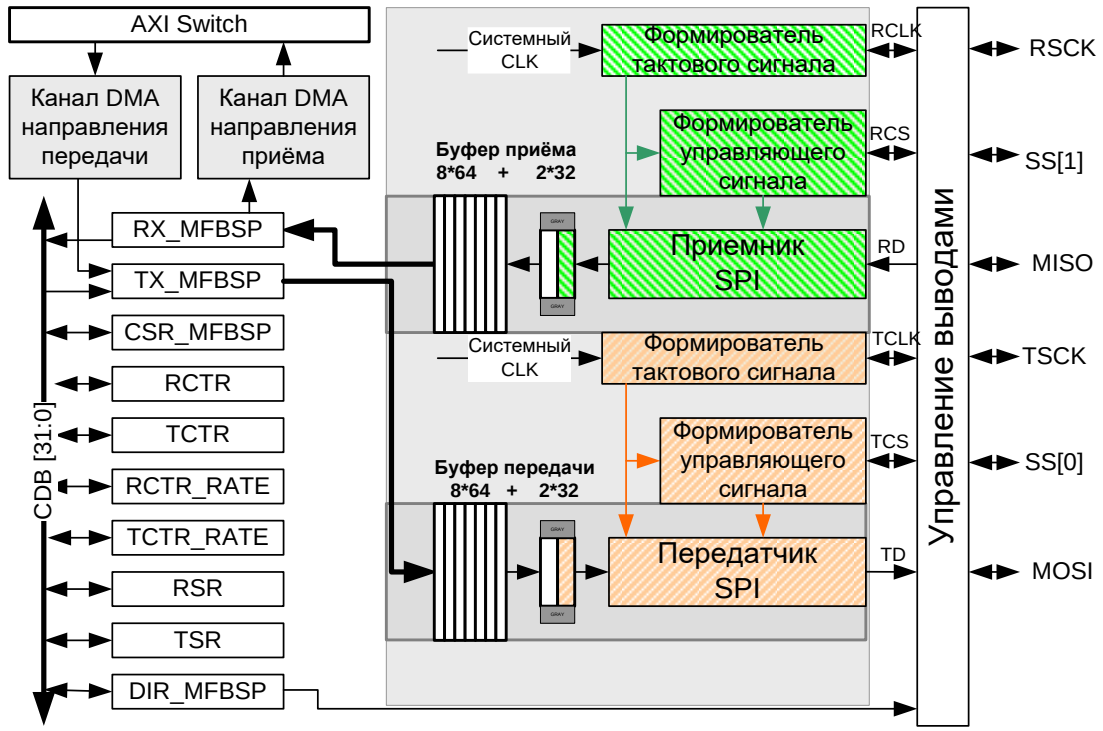


Рисунок 37.22. Структурная схема MFBSP для режима SPI

37.5.15 Варианты соединения порта с внешними устройствами

Программно управляя направлением выводов последовательного порта (см. описание регистра DIR_MFBSP) можно организовать множество вариантов соединения схемы с внешними устройствами через MFBSP (Рисунок 37.23,

Рисунок 37.24,Рисунок 37.25).

MFBSP позволяет подключить два ведомых SPI устройства. Выбор ведомого устройства с которым будет производится обмен осуществляется битами SS, регистра TCTR. Если настройки двух устройств совпадают (для обоих ведомых значения TNEG, TDEL, TWORDLEN одинаковы) тогда перед для смены ведомого устройства достаточно изменить

значение бит SS. Если настройки двух ведомых различаются, то для смены ведомого необходимо сначала выключить MFBSP (spi_i2s_en=0, len=0), и только после этого записать новые настройки в регистры TCTR и RCTR.

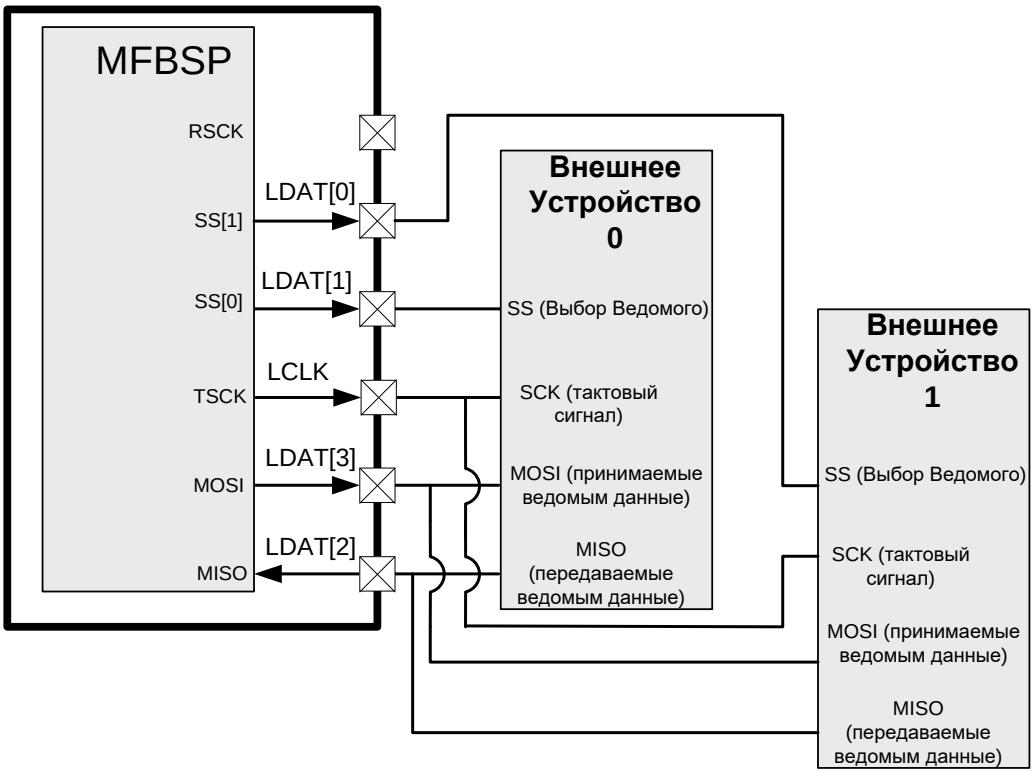


Рисунок 37.23. Подключение к MFBSP двух ведомых устройств по интерфейсу SPI. Приёмник в зависимом от передатчика режиме (режим №6 по Таблица 37.4)



Рисунок 37.24. Подключение MFBSP по интерфейсу SPI к внешнему ведущему устройству. Приёмник в зависимом от передатчика режиме (режим №6 по Таблица 37.4)

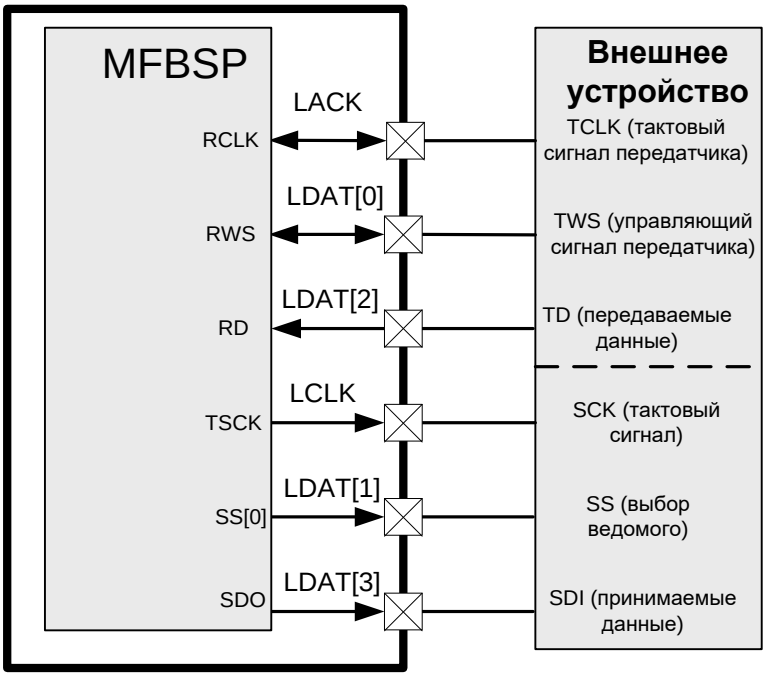


Рисунок 37.25. Организация передачи управляющих данных по интерфейсу SPI и приёма аудиоданных по интерфейсу I2S (режим №5 по Таблица 37.4)

37.5.16 Передача данных в режиме SPI

В режиме SPI возможна передача данных при четырёх сочетаниях бит TDEL и TNEG (Рисунок 37.26,Рисунок 37.27). При этом TNEG – задает начальное состояние вывода TCLK и полярность фронта, по которому производится чтение. TDEL задает смещение передаваемых данных на пол фазы. Значения RNEG и RDEL приёмника должны соответствовать TNEG и TDEL передатчика. После аппаратного сброса SS_DO=0, в этом случае управление сигналом выбора ведомого производится в автоматическом режиме.

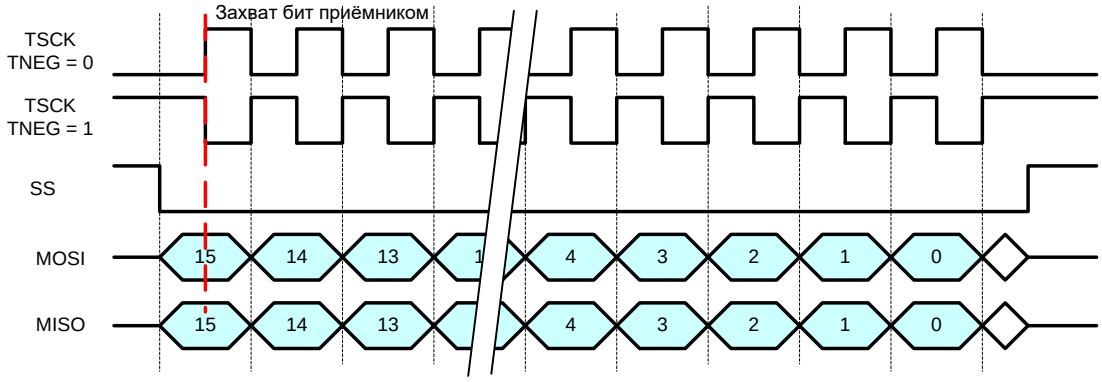


Рисунок 37.26. Передача одного слова в режиме SPI с автоматической генерацией управляющего сигнала TMODE = 1, TMBF = 1, TDEL = 0, SS_DO = 0. Диаграммы тактового сигнала TSCK представлены для различных значений TNEG

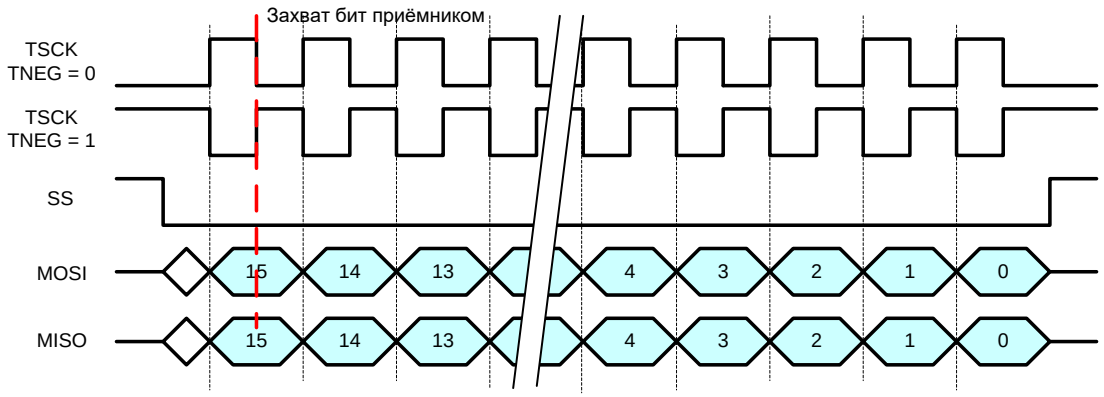


Рисунок 37.27. Передача одного слова в режиме SPI с автоматической генерацией управляющего сигнала TMODE = 1, TMBF = 1, TDEL = 1, SS_DO = 0. Диаграммы тактового сигнала TSCK представлены для различных значений TNEG

Чтобы передать несколько слов без изменения уровня на внешнем выводе SS, можно использовать программное управление внешним выводом SS, в этом случае SS_DO необходимо установить в 1, программно установить вывод SS в 0, записать передаваемые данные в буфер передачи (или включить канал DMA на передачу), дождаться фактического окончания передачи (бит TRUN регистра TSR сбрасывается в 0), после чего программно установить вывод SS в 1 (Рисунок 37.28).

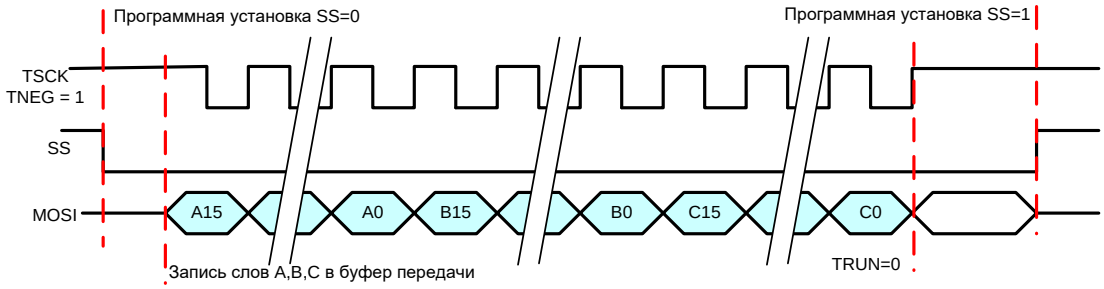


Рисунок 37.28. Передача трёх слов в режиме SPI с программным управлением сигналом SS, TMODE = 1, TMBF = 1, TDEL = 0, TNEG = 0, SS_DO = 1

В режиме SPI также имеется возможность программно регулировать количество слов, которое будет передано без изменения уровня сигнала SS (Рисунок 37.29). Количество слов может быть задано в пределах от 1 до 64 и определятся битом TWORDCNT. Буфер передачи может вместить максимум 18 32-х разрядных слов, если в пределах фрейма передаётся больше 18 слов необходимо следить за тем, чтобы буфер передачи был не пуст (при включенном канале DMA это происходит автоматически).

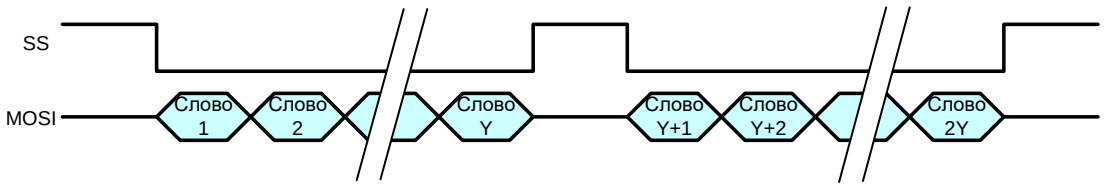


Рисунок 37.29. Передача в режиме SPI, TWORDCNT=Y-1

В режиме ведомого устройства сигнал выбора ведомого предварительно пересинхронизируется на внутреннюю частоту порта, поэтому для устойчивой работы порта в режиме ведомого SPI устройства уровень сигнала SS, если необходима его установка в 1 между передачами, должен удерживаться как минимум два периода внутренней частоты CLK.

Непосредственно к тактовому сигналу TSCK данное ограничение не применяется, т.е. частота TSCK может быть больше CLK.

Когда MFBSP работает в режиме ведущего SPI устройства, время удержания сигнала SS при автоматическом формировании данного сигнала может регулироваться программно. В этом случае время между последним фронтом тактового сигнала для последней пересылки и установкой сигнала SS в 1 равно времени между установкой и сбросом сигнала SS и равно времени между сбросом сигнала SS первым фронтом тактового сигнала для новой пересылки. Это время определятся как TSS = (TSS_RATE+1)*TTCLK/2, где TTCLK – период тактового сигнала, генерируемого портом для последовательной передачи данных. Если необходимо формировать сигнал SS средствами приёмника – то для этих целей используется поле RSS_RATE (Рисунок 37.30).

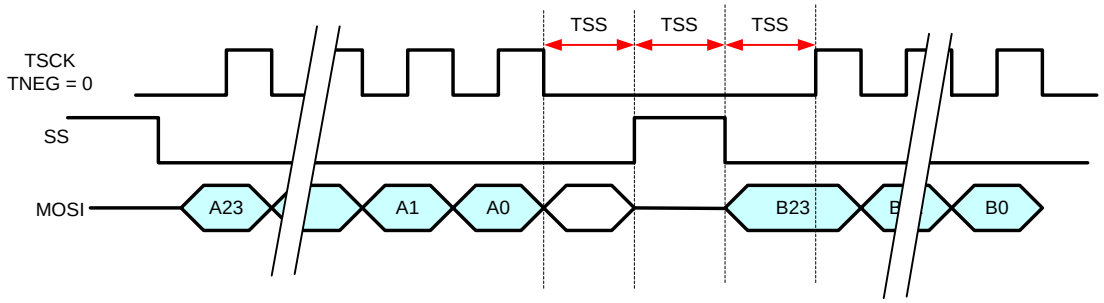


Рисунок 37.30. Управление временем удержания сигнала SS в высоком уровне между передачами, на картинке TNEG = 0, TDEL = 0, TMBF = 1, TWORDLEN = 23, TSS_RATE = 1

37.5.17 Пример чтения 8-разрядного слова по заданному адресу из ведомого устройства с интерфейсом C-BUS

Для чтения слова по указанному адресу по интерфейсу C-BUS необходима передача двух 8ми битных слов.

Для организации такого чтения необходимо записать соответствующий ведомому устройству бит SS, регистра TCTR, 1;

Перевести порт в режим SPI (LEN = 0, SPI_I2S_EN = 1, RMODE = 1, TMODE = 1);

Настроить приемник и передатчик: TDEL = RDEL = 0; TNEG = RNEG = 0; TWORDLEN = RWORDLEN = 5’h0F; RCLK_CP = 1; RCS_CP =1, SS_DO = 0;

Включить приемник и передатчик REN = 1, TEN = 1;

Записать в регистр TX_MFBSP 32-х разрядное слово, содержащее во втором байте 7ми разрядный адрес и бит WR, значение младшего байта не важно.

Ожидаем до тех пор, пока в буфер приёма не будет записано приятое слово (RSR[0] сбрасывается в 0)

В прочитанном по адресу RX_MFBSP 32-х разрядном слове, младшие 8 бит – слово, прочитанное из ведомого устройства.

На Рисунок 37.31 представлены временные диаграммы для передачи по интерфейсу CBUS.

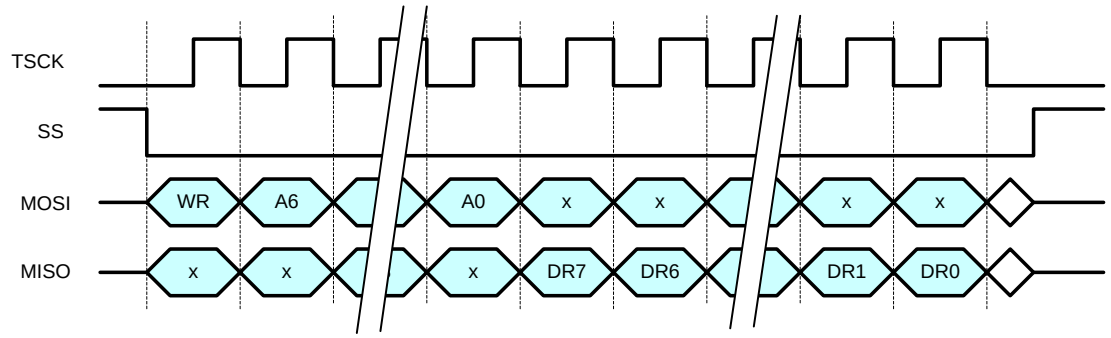


Рисунок 37.31. Пример чтения 8-ми разрядного слова из ведомого устройства (интерфейс C-BUS)

37.5.18 Формирование тактовых сигналов приёмника (RSCK) и передатчика (TSCK)

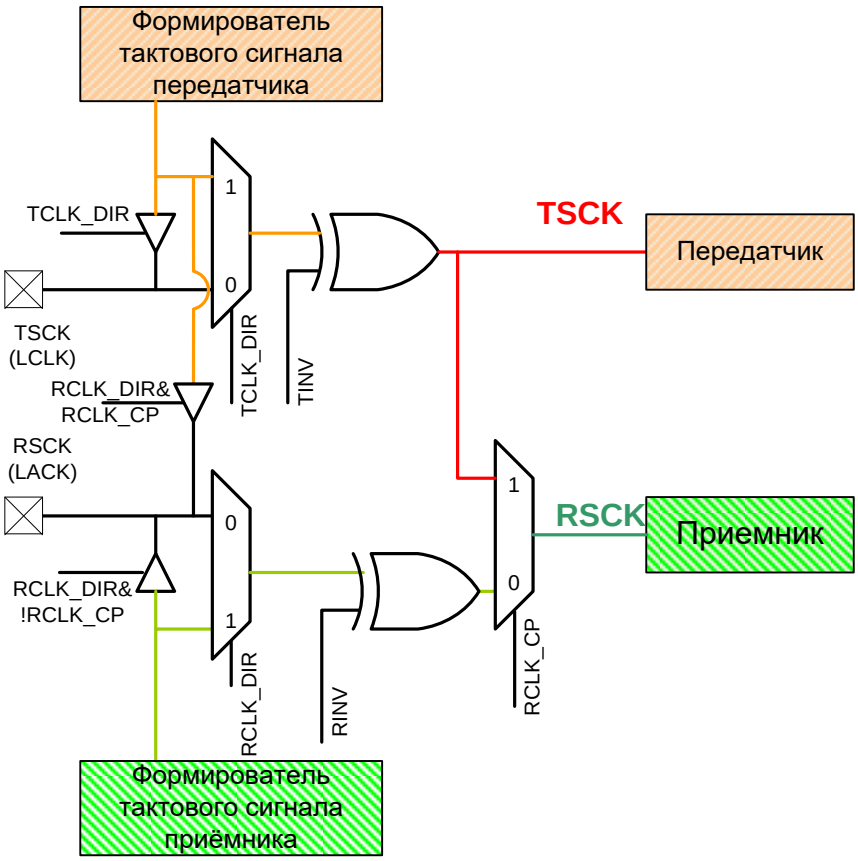


Рисунок 37.32. Схема формирования тактовых сигналов приёмника и передатчика в режиме SPI

На Рисунок 37.32 представлена схема формирования тактовых сигналов приёмника и передатчика в режиме SPI.

В зависимости от значения бита TCLK_DIR, тактовый сигнал передатчика TSCK может как формироваться самим передатчиком, так приниматься с внешнего вывода. В зависимости от значений бит TMODE, TNEG и TDEL тактовый сигнал либо передаётся передатчику без изменений, либо инвертируется.

В зависимости от значения бита RCLK_DIR, тактовый сигнал приёмника RSCK может как формироваться самим приёмником, так приниматься с внешнего вывода. В зависимости от значений бит RMODE, RNEG и RDEL тактовый сигнал либо передаётся приёмнику без изменений, либо инвертируется.

Если бит RCLK_CP установлен в 1, то тактовый сигнал приёмника копирует тактовый сигнал передатчика. Для корректной работы устройства в этом случае настройки полярности тактового сигнала приёмника и передатчика должны совпадать (TNEG=RNEG, TDEL=RDEL).

При RCLK_CP = 1 тактовый сигнал передатчика передаётся на внешний вывод приёмника, только если передатчик сам формирует тактовый сигнал и вывод тактового сигнала приёмника сконфигурирован как выход (TCLK_DIR=1, RCLK_DIR=1).

37.5.19 Формирование управляющих сигналов приёмника и передатчика в режиме SPI

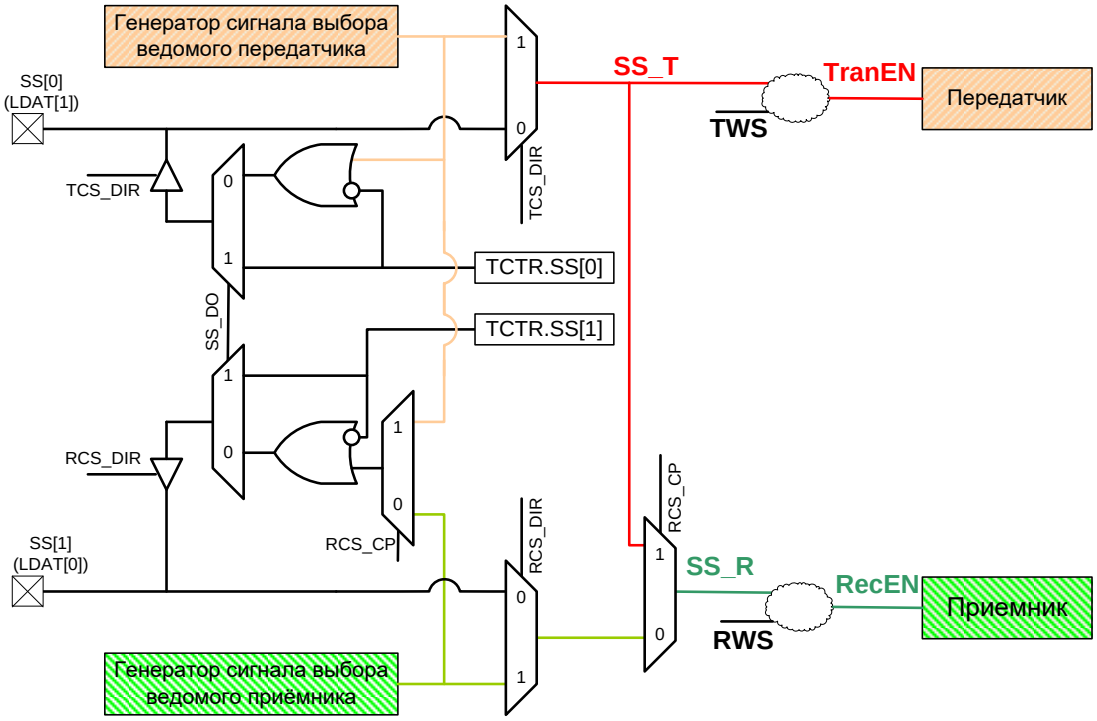


Рисунок 37.33. Схема формирования управляющих сигналов в режиме SPI

На Рисунок 37.33 представлена схема формирования управляющих сигналов в режиме SPI.

SS – шина выбора ведомого устройства. Низкий уровень сигнала SS, поданный на ведомое устройство означает, что данное устройство выбрано и с приходом тактового сигнала SCK должно начать обмен данными с ведущим устройством.

MFBSP с зависимым от передатчика приёмником в режиме ведущего позволяет параллельно подключать до двух ведомых устройств по шине SPI и формировать сигналы выбора ведомого устройства как в автоматическом режиме, так и программно.

MFBSP с зависимым от передатчика приёмником может работать как ведомое SPI устройство, управляемое внешним сигналом SS[0] и внешней тактовой частотой TSCK, обеспечивая обмен данными в дуплексном режиме.

MFBSP позволяет организовать независимый приём и передачу данных по интерфейсу SPI. В этом случае SS[0] – управляющий сигнал передатчика, SS[1] – управляющий сигнал приёмника.

При TCS_DIR = 1 передатчик SPI формирует сигнал выбора ведомого, SS[0] - выход. В автоматическом (SS_DO=0) режиме формирования управляющего сигнала перед началом передачи очередного слова сигнал выбора ведомого переводится в низкий уровень, а по окончании передачи слова сигнал выбора ведомого снова переводится в высокий уровень. Изменение уровня на выводе SS[0] происходит только в случае, если соответствующий бит SS[0] регистра TCTR установлен в 1. Если приёмник в зависимом от передатчика режиме (RCS_CP = 1) и SS[1] сконфигурирован как выход (RCS_DIR=1), то вывод SS[1] используется как сигнал выбора дополнительного ведомого устройства. Изменение уровня на выводе SS[1] происходит только, в случае, если соответствующий бит SS[1] регистра TCTR установлен в 1. В случае программного управления шиной SS (SS_DO = 1) значения бит SS[1:0] контрольного регистра TCTR передаются непосредственно на выводы SS[1:0].

Если приёмник в зависимом от передатчика режиме (RCS_CP=1) и вывод SS[0] сконфигурирован как вход (TCS_DIR = 0), тогда MFBSP работает в режиме дуплексного ведомого SPI устройства. Сигнал выбора ведомого принимается с внешнего вывода SS[0] и используется как приёмником, так и передатчиком.

Если приёмник работает в независимом от передатчика режиме (RCS_CP=0), то в режиме ведущего, когда вывод SS[1] сконфигурирован как выход (RCS_DIR=1) формируемый приёмником сигнал выбора ведомого направляется на вывод SS[1]. При автоматическом формировании управляющего сигнала (SS_DO = 0) перед началом приёма очередного слова сигнал SS[1] автоматически переводится в низкий уровень и переводится в высокий уровень по окончании приёма каждого слова. В режиме ведущего устройства приём слов приёмником ведется до заполнения буфера приёма. В режиме ведомого устройства, когда вывод SS[1] сконфигурирован как вход (RCS_DIR=0) независимый приёмник (RCS_CP=0) принимает сигнал выбора ведомого с вывода SS[1].

В режиме SPI направление выводов тактового сигнала и управляющего сигнала должно строго совпадать. Т.е. TCLK_DIR=TCS_DIR. В случае если приёмник работает независимо от передатчика, то RCLK_DIR=RCS_DIR.

37.5.20 Тракт передачи данных

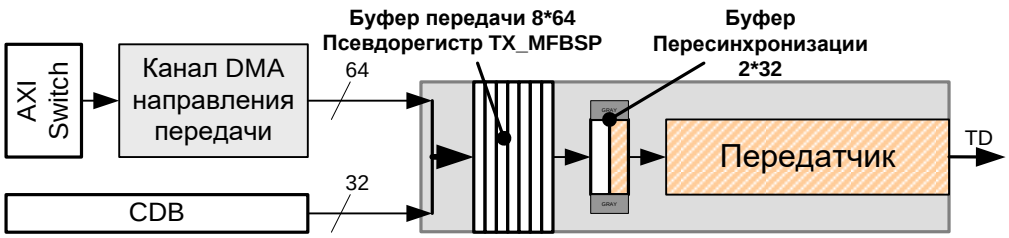


Рисунок 37.34. Тракт передачи данных в режиме SPI

На Рисунок 37.34 представлен тракт передачи данных в режиме SPI.

Чтобы инициировать передачу данных по последовательному порту, необходимо включить последовательный порт (SPI_I2S_EN=1) и передатчик (TEN=1), после чего либо начать производить запись передаваемых 32-х разрядных слов в буфер передачи по адресу псевдорегистра TX_MFBSP, либо включить канал DMA в направления передачи для соответствующего порта (в этом случае обмен данными с портом будет вестись 64-х разрядными словами).

Данные записанные в буфер передачи автоматически перемещаются в буфер пересинхронизации направления передачи, если он не полон. Запись в буфер пересинхронизации направления передачи осуществляется на системной частоте CLK, чтение из буфера пересинхронизации осуществляется на частоте передатчика TCLK. Как только в буфере пересинхронизации оказалось хотя бы одно слово передатчиком инициируется передача. Передатчиком производится последовательная выдача бит очередного 32-х разрядного слова до тех пор, пока число переданных бит не достигнет TWORDLEN+1, после чего производится считывание очередного слова из буфера пересинхронизации. По мере передачи слов в освобождающийся буфер пересинхронизации перемещается слово из буфера передачи. После выборки последнего слова из буфера передачи (буфер передачи пуст) в буфере пересинхронизации остаётся еще два слова. Фактическое окончание передачи можно идентифицировать по состоянию буфера пересинхронизации, либо считав бит TRUN регистра TSR.

Если управляющий сигнал формируется передатчиком, то при считывании последнего слова из буфера пересинхронизации передача останавливается. Передача продолжится только после того как в буфер пересинхронизации снова начнут поступать данные.

Если передатчик использует внешнюю частоту и внешний управляющий сигнал, в целях экономии мощности системная частота может быть установлена меньшей, чем внешняя частота передатчика, однако ее должно быть достаточно для того, что бы успеть переместить очередное слово в буфер пересинхронизации (за время передачи одного слова должно быть хотя бы три импульса системной частоты CLK). Если внешний управляющий сигнал инициировал передачу слова при пустом буфере пересинхронизации, устанавливается флаг ошибки передачи (TERR), в этом случае передаётся ошибочное слово. Если управляющий сигнал формируется самим передатчиком, системная частота может быть много меньше частоты передатчика, однако это скажется на скорости передачи данных.

Установка бита TERR в процессе передачи говорит о том, что порт произвел попытку чтения из пустого буфера передачи. Это значит, что передатчиком было передано некорректное слово.

В направлении передачи порт обладает буферизацией на 18 32-х разрядных слов. В случае передачи данных посредством DMA запись блоков данных в буфер передачи происходит

до тех пор, пока буфер готов принять очередной блок, размер которого определяется битами WN, регистра CSR соответствующего канала DMA.

Степень заполнения буфера можно программно регулировать, используя биты TBES. В этом случае значение выражения TBES+1 – задает эффективный размер буфера передачи. Передача очередной пачки происходит только в случае, если при записи этой пачки в буфер передачи число 64 разрядных слов в буфере передачи не превысит TBES+1. При попытке передать пачку со значением WN > TBES, значение WN автоматически корректируется (большая пачка передается в несколько этапов пачками меньшего размера).

Так, при WN=0 и TBES=0 очередное 64-х разрядное слово будет подкачиваться в буфер передачи только если он пуст. В этом случае по окончании работы DMA в момент прерывания от соответствующего канала передатчику останется передать ещё 4 32-х разрядных слова (два 32-х разрядных слова в буфере пересинхронизации и одно 64-х разрядное слово в буфере передачи).

Установка бита SPI_I2S_EN в 0 приведет к программному сбросу передатчика, и все данные находящиеся в буфере передачи будут утеряны.

37.5.21 Тракт приёма данных

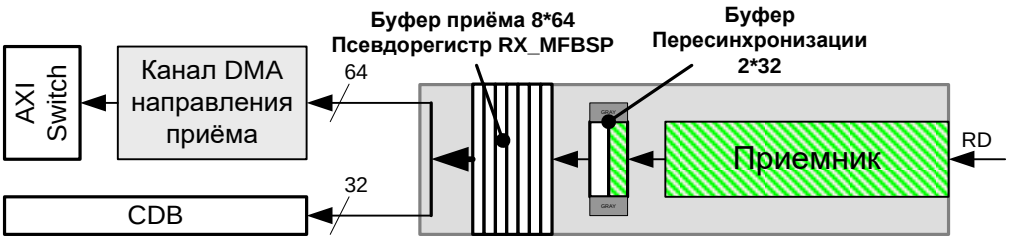


Рисунок 37.35. Тракт приёма данных в режиме SPI

На Рисунок 37.35 представлен тракт приёма данных в режиме SPI.

Что бы перевести приёмник в режим готовности необходимо включить последовательный порт (SPI_I2S_EN=1) и приёмник (REN=1), после чего либо начать ожидание появления прочитанных данных в буфере приёма, либо включить канал DMA в направления приёма для соответствующего порта.

Приёмник принимает последовательные биты, поступающие с внешнего вывода до тех пор, пока число принятых бит не достигнет значения RWORDLEN+1. После этого принятое 32-х разрядное слово (если RWORDLEN<31 незадействованные биты обнуляются) перемещается в буфер пересинхронизации. Запись в буфер пересинхронизации направления приёма осуществляется на частоте приёмника RCLK, чтение из буфера пересинхронизации осуществляется на системной частоте CLK. Из буфера пересинхронизации принятое слово автоматически перемещается в буфер приёма, если он не полон. Если в буфере приёма есть хотя бы одно 32-х разрядное слово, то принятые 32-х

разрядные слова можно считывать, обращаясь по адресу псевдорегистра RX_MFBSP. Принимать данные можно также включив соответствующий порту канал DMA направления приёма (в этом случае обмен данными с портом осуществляется 64-х разрядными словами).

Если приёмник использует внешнюю частоту, то в целях экономии мощности системная частота может быть установлена меньшей, чем внешняя частота приёмника, однако ее должно быть достаточно для того, чтобы успеть переместить очередное слово из буфера пересинхронизации (за время приёма одного слова должно быть хотя бы три импульса системной частоты CLK). Если при заполненном буфере пересинхронизации приёмником был произведен приём очередного слова и инициирована попытка записи в буфер пересинхронизации, устанавливается флаг ошибки приёма (RERR), а последнее принятое слово теряется.

Установка бита RERR в процессе передачи говорит о том, что порт произвел попытку записи в полный буфер приёма. Это значит, что принятое слово было потеряно.

В направлении приёма порт обладает буферизацией на 18 32-х разрядных слов. В случае приёма данных посредством DMA чтение блоков данных из буфера приёма происходит до тех пор, пока в буфере приёма достаточно слов для чтения очередного блока, размер которого определяется битами WN, регистра CSR соответствующего канала DMA. DMA обмены возможны только 64 разрядными словами, таким образом, если было принято нечетное количество 32-х разрядных слов, после окончания работы DMA необходимо прочитать оставшееся слово, обратившись к псевдорегистру RX_MFBSP.

Установка бита SPI_I2S_EN в 0 приведет к программному сбросу приёмника и все данные находящиеся в буфере приёма будут утеряны.

37.5.22 Прерывания от последовательного порта

Прерывание MFBSP_RXBUF устанавливается, в случае если включен приемник (I2S_SPI_EN=1, REN = 1) и в буфер приёма записано количество слов большее, чем установлено уровнем прерывания RLEV, либо произошла ошибка приема (RERR = 1).

Прерывание MFBSP_TXBUF устанавливается, в случае если включен передатчик (I2S_SPI_EN=1, REN = 1) и в буфере передачи осталось количество слов меньшее, либо равное чем установлено уровнем прерывания TLEV, либо произошла ошибка передачи (TERR = 1).

37.6 Работа MFBSP в режиме линкового порта (LPORT)

37.6.1 Назначение линкового порта

Линковый порт предназначен для обмена данными между различными микросхемами последовательно-параллельным кодом.

Порт может передавать 32-х разрядные слова частями по 4 бита за 8 пересылок, либо частями по 8 бит за 4 пересылки, выбор одного из этих режимов осуществляется установкой бита LDW, регистра CSR_MFBSP.

37.6.2 Регистр управления и состояния CSR_MFBSP (режим LPORT)

Таблица 37.33. Назначение разрядов регистра CSR_MFBSP в режиме LPORT

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:17	-	Резерв	RW	0
16	-	Резерв	RW	0
15	-	Резерв	RW	0
14:11	LCLK_RATE [4:1]	Делитель частоты LPORT: LCLK = CLK/(2*(LCLK_RATE+1))	RW	0
10	-	Резерв	RW	0
9	SPI_I2S_EN	В режиме LPORT должен быть установлен в 0	RW	0
8	SRQ_RX	Признак запроса обслуживания на прием данных	R	0
7	SRQ_TX	Признак запроса обслуживания на передачу данных	R	0
6	LDW	Разрядность внешней шины данных: 0 - 4-разряда (32-разрядное слово передается за 8 посылок); 1 - 8-разряда (32-разрядное слово передается за 4 посылки).	RW	0
5	LRERR	Ошибка приема данных: 0 – приняты все биты данных; 1 – приняты не все биты данных.	R	0
4:3	LSTAT	Состояние буфера: При LTRAN = 0 показывает состояние буфера приёма При LTRAN = 1 показывает состояние буфера передачи 00 – буфер пуст; 10 – буфер не пуст; 11 – буфер полон.	R	0
2	LCLK_RATE[0]	Делитель частоты LPORT: LCLK = CLK/(2*(LCLK_RATE+1))	RW	0
1	LTRAN	Режим работы порта: 0 – приемник; 1 – передатчик.	RW	0
0	LEN	Разрешение работы порта: 0 – все выводы порта находятся в высокоимпедансном состоянии; 1 – порт работает в соответствии с состоянием бита LTRAN.	RW	0

Биты LSTAT, LRERR сбрасываются при LEN=0.

37.6.3 Регистр состояния приёмника RSR (режим LPORT)

Таблица 37.34. Назначение разрядов регистра RSR в режиме LPORT

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:28	-	Резерв	-	0
27:24	RB_DIFF	Количество принятых 64-разрядных слов в буфере приёма (мах 8).	R	0
23:19	-	Резерв	-	0
18:16	RLEV	Порог прерывания от буфера приёма: Прерывание формируется если число принятых 64-х разрядных слов больше RLEV	RW	7
15:11	-	Резерв	-	0
10	RXBUF	Результирующее прерывание MFBSP_RXBUF	R	0
9	RXBUF_D	Прерывание MFBSP_RXBUF без механизма автоматического сброса при чтении RSR	R	0
8	RXBUF_R	Прерывание MFBSP_RXBUF с механизмом автоматического сброса при чтении RSR	R	0
7	-	В режиме LPORT не используется	R	0
6	-	В режиме LPORT не используется	R	0
5	RSBF	Буфер пересинхронизации в направлении приёма полон: 0 – буфер пересинхронизации в направлении приёма не полон 1 – буфер пересинхронизации в направлении приёма полон	R	0
4	RSBE	Буфер пересинхронизации в направлении приёма пуст: 0 – буфер пересинхронизации в направлении приёма не пуст 1 – буфер пересинхронизации в направлении приёма пуст	R	1
3	RBHL	Достигнут порог прерывания в буфере приёма: 1 – число 64-х разрядных слов в буфере приёма больше чем задано в RLEV 0 – число 64-х разрядных слов в буфере приёма меньше либо равно RLEV	R	0
2	RBHF	Буфер приёма полон на половину или более: 1 – буфер приёма заполнен на половину или больше (из буфера приёма можно считать как минимум 4 слова) 0 – буфер приёма заполнен меньше, чем наполовину	R	0
1	RBF	Буфер приёма полон: 0 – буфер приёма не полон 1 – буфер приёма полон	R	0
0	RBE	Буфер приёма пуст: 0 – буфер приёма не пуст 1 – буфер приёма пуст	R	1

37.6.4 Регистр состояния передатчика TSR (режим LPORT)

Таблица 37.35. Назначение разрядов регистра TSR в режиме LPORT

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:28	-	резерв	-	0

27:24	TB_DIFF	Количество свободных 64-разрядных позиций в буфере передачи (в буфер передачи можно записать еще TB_DIFF 64-разрядных слов).	R	8
23	-	Резерв	-	0
22:20	TBES	Эффективный размер буфера передачи Актуален только для режима работы с DMA. Значение TBES+1 – определяет максимальный объем буфера передачи. Т.е. в режиме работы с DMA буфер передачи не может быть заполнен больше, чем на TBES+1 64 разрядных слов.	RW	7
19	-	Резерв	-	0
18:16	TLEV	Порог прерывания от буфера передачи: Прерывание формируется если число 64-х разрядных слов в буфере передачи меньше либо равно TLEV. В режиме передачи данных с использованием DMA определяет степень заполнения буфера передачи, при которой происходит запись в буфер очередной пачки данных	RW	0
15:11	-	Резерв	-	0
10	TXBUF	Результирующее прерывание MFBSP_TXBUF	R	0
9	TXBUF_D	Прерывание MFBSP_TXBUF без механизма автоматического сброса при чтении TSR	R	0
8	TXBUF_R	Прерывание MFBSP_TXBUF с механизмом автоматического сброса при чтении TSR	R	0
7	-	В режиме LPORT не используется	R	0
6	-	В режиме LPORT не используется	RW	0
5	-	В режиме LPORT не используется	R	0
4	-	В режиме LPORT не используется	R	0
3	TBLL	Достигнут порог прерывания в буфере передачи: 1 – число 64-х разрядных слов в буфере передачи меньше либо равно TLEV 0 – число 64-х разрядных слов в буфере передачи больше TLEV	R	1
2	TBHF	Буфер передачи заполнен на половину или более: 1 – буфер передачи заполнен на половину или больше 0 – буфер передачи заполнен меньше чем на половину (в буфер передачи можно записать еще как минимум 4 слова)	R	0
1	TBF	Буфер передачи полон: 0 – буфер передачи не полон 1 – буфер передачи полон	R	0
0	TBE	Буфер передачи пуст: 0 – буфер передачи не пуст 1 – буфер передачи пуст	R	1
31:15	-	Резерв	-	0
14	RXBUF_R_EN	Разрешение автоматического сброса прерывания MFBSP_RXBUF 0 – прерывание не сбрасывается при чтении RSR 1 – прерывание сбрасывается при чтении RSR	RW	1
13	RX_LEV_IRQ_EN	Разрешение прерывания по уровню заполнения буфера приема 0 – прерывание MFBSP_RXBUF не будет устанавливаться при превышении порога RLEV 1 - прерывание MFBSP_RXBUF будет устанавливаться при превышении порога RLEV	RW	1

12	RX_ERR_IRQ_EN	Разрешение прерывания при переполнении буфера приема 0 – прерывание MFBSP_RXBUF не будет устанавливаться при переполнении буфера приема 1 - прерывание MFBSP_RXBUF будет устанавливаться при переполнении буфера приема	RW	1
11:7	-	Резерв	-	0
6	TXBUF_R_EN	Разрешение автоматического сброса прерывания MFBSP_TXBUF 0 – прерывание не сбрасывается при чтении TSR 1 – прерывание сбрасывается при чтении TSR	RW	1
5	TX_LEV_IRQ_EN	Разрешение прерывания по уровню заполнения буфера передачи 0 – прерывание MFBSP_TXBUF не будет устанавливаться, если число слов в буфере передачи меньше порога TLEV 1 - прерывание MFBSP_TXBUF будет устанавливаться, если число слов в буфере передачи меньше порога TLEV	RW	1
4	TX_ERR_IRQ_EN	Разрешение прерывания при переполнении буфера приема 0 – прерывание MFBSP_TXBUF не будет устанавливаться при чтении из пустого буфера передачи 1 - прерывание MFBSP_TXBUF будет устанавливаться при чтении из пустого буфера передачи	RW	1
3:1	-	Резерв	-	0
0	LPT_IRQ_EN	Разрешение прерывания по запросу на обслуживание 0 – SRQ запрещено 1 – SRQ разрешено	RW	1

37.6.5 Регистр аварийного управления портом EMERG_MFBSP (режим LPORT)

Таблица 37.36. Назначение разрядов регистра EMERG_MFBSP в режиме SPI

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:6	-	Резерв	-	0
5	RX_DBG	Включение аварийной прокачки данных канала DMA направления приема. 0 – работа в обычном режиме 1 – сигнал готовности передачи данных по DMA каналу направления приема постоянно удерживается в 1 (по DMA принимаются невалидные данные)	RW	0
4	TX_DBG	Включение аварийной прокачки данных канала DMA направления передачи. 0 – работа в обычном режиме 1 – сигнал готовности передачи данных по DMA каналу направления передачи постоянно удерживается в 1 (данные передаваемые DMA теряются)	RW	0
3	-	Резерв	-	0
2	RST_RXBUF	Сброс буфера приема последовательного порта и буфера пересинхронизации направления приема. Запись единицы в данный разряд приводит к сбросу буфера, после чего данный бит снова устанавливается в 0.	RW	0

1	RST_TXBUF	Сброс буфера передачи последовательного порта и буфера пересинхронизации направления передачи. Запись единицы в данный разряд приводит к сбросу буфера, после чего данный бит снова устанавливается в 0.	RW	0
0	RST_LPTBUF	Сброс буфера линкового порта и буфера пересинхронизации направления приёма. Запись единицы в данный разряд приводит к сбросу буфера, после чего данный бит снова устанавливается в 0.	RW	0

37.6.6 Регистр маски прерываний от порта IMASK (режим LPORT)

Таблица 37.37. Назначение разрядов регистра IMASK в режиме SPI

37.6.7 Структурная схема MFBSP для режима линкового порта

На Рисунок 37.36 представлена структурная схема MFBSP для режима линкового порта.

Включение линкового порта происходит при установке бита LEN в 1 и бита SPI_I2S_EN в 0.

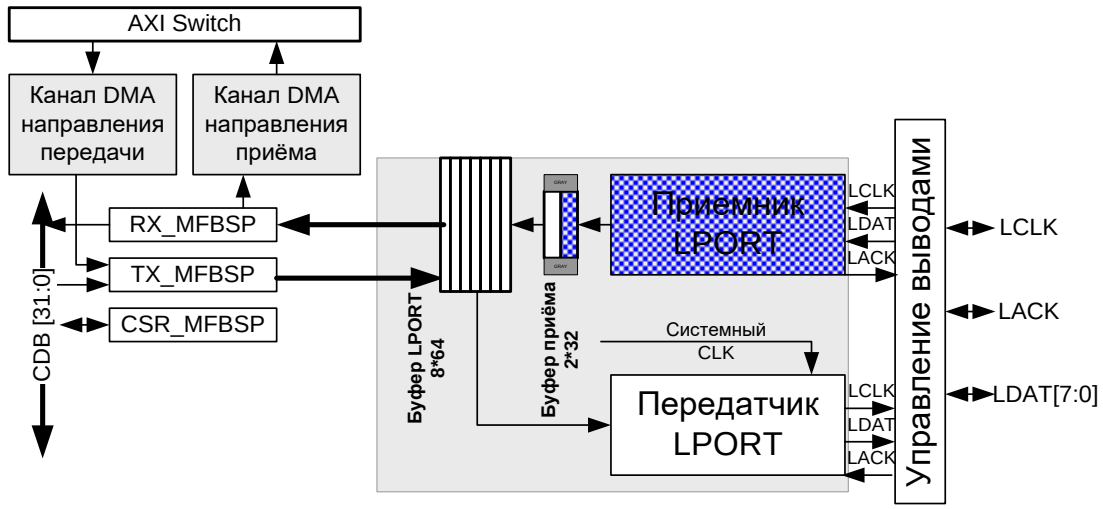


Рисунок 37.36. Структурная схема MFBSP для режима LPORT

37.6.8 Соединение с внешними устройствами

На Рисунок 37.37 и Рисунок 37.38 представлены варианты соединения MFBSP с внешними устройствами в режиме линкового порта.

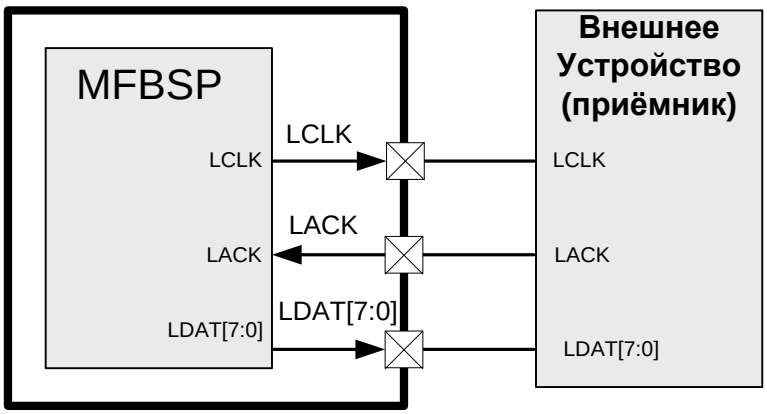


Рисунок 37.37. MFBSP в режиме передатчика LPORT (LCLK, LDAT-выходы, LACK - вход) (режим №2 по Таблица 37.4)

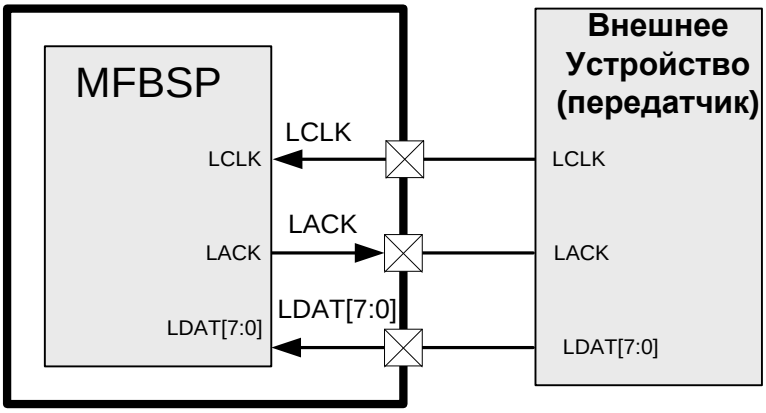


Рисунок 37.38. MFBSP в режиме приёмника LPORT (LCLK, LDAT-входы, LACK - выход) (режим №2 по Таблица 37.4)

37.6.9 Передача данных по линковому порту

По линковому порту передача данных происходит в одном направлении (либо передача данных, либо приём данных).

Для смены направления обмена данными по линковому порту необходимо сначала выключить порт (установить бит LEN, регистра CSR_MFBSP в 0), затем включить порт, установив требуемое значение направления передачи данных (бит LTRAN, регистра CSR_MFBSP)

Передача данных по линковому порту возможна для любых сочетаний частот приёмника и передатчика, скорость передачи данных будет определяться самым медленным устройством.

Для корректной передачи данных необходимо, чтобы значение бита LDW у приёмника и у передатчика совпадало.

Если для передатчика LDW=1, а для приёмника LDW=0 приёмник будет упаковывать два 32х разрядных слова в одно 32-х разрядное слово, выкидывая из каждого байта старшие 4 бита.

Установка значений LDW для передатчика LDW=0, а для приёмника LDW=1 не допускается.

Временная диаграмма работы линкового порта приведена на Рисунок 37.39.

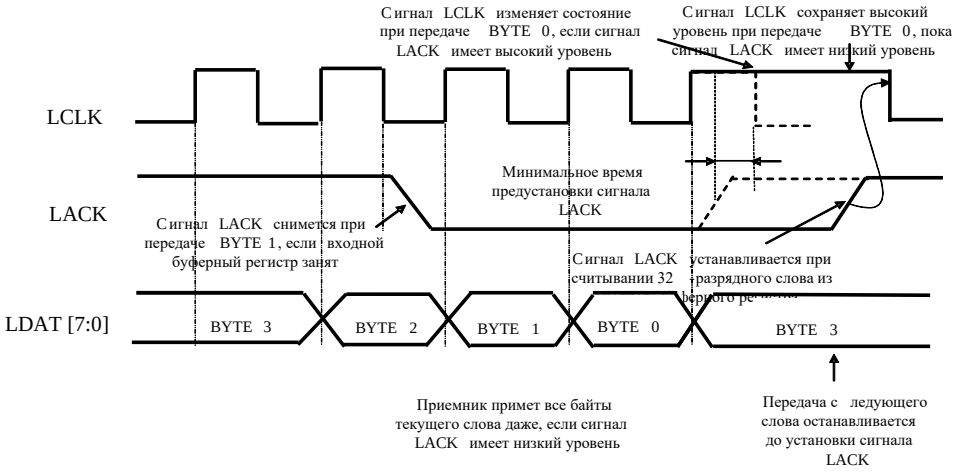


Рисунок 37.39. Временная диаграмма работы линкового порта (LDW=1)

При LDW=0 передача 32-разрядного слова выполняется за 8 посылок, а при LDW=1 - за 4 посылки. Передатчик изменяет данные LDAT по положительному фронту LCLK, а приемник защелкивает данные в буфере приёма по отрицательному фронту.

Исходное состояние сигнала LACK – высокий уровень. Сигнал LACK снимется приемником по заднему фронту LCLK при передаче BYTE1, если в буфере приёма осталось место для приёма всего одного слова. При этом приемник примет все байты текущего 32-разрядного слова даже, если сигнал LACK имеет низкий уровень. Сигнал LACK устанавливается при считывании 32-разрядного слова из входного буферного регистра.

Передатчик после выставления BYTE0 анализирует состояние сигнала LACK. Если LACK=1, то LCLK продолжает изменять свое состояние и после BYTE 0 передается BYTE 1 и так далее. Если LACK=0, то LCLK сохраняет высокий уровень при передаче BYTE 0, пока сигнал LACK имеет низкий уровень.

Если линковый порт деактивизирован (LEN=0) сигналы LDAT, LCLK LACK являются входами. Поэтому эти сигналы необходимо привязывать к земле через резисторы 10 кОм. Если порт настроен как передатчик, LDAT и LCLK становятся выходами, а LACK – входом. Если порт настроен как приемник, LDAT и LCLK становятся входами, а LACK – выходом.

LPORT может выполнять либо только приём, либо только передачу данных. Поэтому LPORT снабжен одним буфером на 8 64-х разрядных слов, используемом как в направлении

приёма, так и в направлении передачи. В направлении приёма дополнительно встроен буфер на 2 32-х разрядных слова, используемый для пересинхронизации с внешней частоты LCLK на внутреннюю системную частоту.

Таким образом, LPORT обладает буферизацией в направлении передачи на 8 64-разрядных слов (16 32-разрядных слов) и буферизацией в направлении приёма на 9 64-разрядных слов (18 32-разрядных слов).

В случае передачи данных посредством DMA запись блоков данных в буфер передачи происходит до тех пор, пока буфер готов принять очередной блок, размер которого определяется битами WN, регистра CSR соответствующего канала DMA.

Степень заполнения буфера можно программно регулировать, используя биты TBES. В этом случае значение выражения TBES+1 – задает эффективный размер буфера передачи. Передача очередной пачки происходит только в случае, если при записи этой пачки в буфер передачи число 64 разрядных слов в буфере передачи не превысит TBES+1. При попытке передать пачку со значением WN > TBES, значение WN автоматически корректируется (большая пачка передается в несколько этапов пачками меньшего размера).

Так, при WN=0 и TBES=0 очередное 64-х разрядное слово будет подкачиваться в буфер передачи только если он пуст. В этом случае по окончании работы DMA в момент прерывания от соответствующего канала передатчику останется передать ещё 4 32-х разрядных слова (два 32-х разрядных слова в буфере пересинхронизации и одно 64-х разрядное слово в буфере передачи).

Принимаемые портом данные сначала помещаются в буфер пересинхронизации и только через два такта перемещаются в буфер LPORT. При опросе контрольных регистров порта доступно состояние только буфера LPORT без учёта буфера пересинхронизации. Таким образом, после заполнения основного буфера LPORT могут быть приняты ещё два 32-х разрядных слова, которые будут перемещаться из буфера пересинхронизации в общий буфер LPORT по мере освобождения буфера LPORT.

Запись данных в буфер пересинхронизации LPORT осуществляется по внешней частоте LCLK, а перемещение данных из буфера пересинхронизации в буфер LPORT осуществляется по внутренней системной частоте CLK. Если внутренняя системная частота более чем в 4 раза меньше внешней частоты LCLK, скорости перемещения данных между двумя буферами может быть недостаточно, что будет приводить к периодическому заполнению буфера пересинхронизации. К потере данных это не приведет, поскольку в LPORT предусмотрен механизм останова передачи по заполнению буфера приёма, однако это приведёт к замедлению обмена данными по линковым портам.

37.6.10 Прерывания от линковых портов

Если линковый порт не активизирован (LEN=0, SPI_I2S_EN=0), он формирует прерывание по запросу обслуживания, если:

- на внешней шине выставлены данные на прием (активное состояние сигнала LCLK);
- из внешней шины поступил запрос на выдачу данных (активное состояние сигнала LACK).

Данное прерывание сбрасывается после установки LEN=1.

Если MFBSP используется в режиме линкового порта, то чтобы избежать ложной установки прерывания SRQ в случае, когда порт выключен и на выводах LACK или LCLK установлено высокоимпедансное состояние, необходимо к выводам LACK и LCLK подключить pull-down резисторы.

При LPT_IRQ_EN=0 данное прерывание маскируется

Если включен линковый порт (LEN=1), прерывания от MFBSP формируются, в случае если в буфер приёма записано количество слов большее, чем установлено уровнем прерывания RLEV (MFBSP_RXBUF), либо если при включенном передатчике в буфере передачи осталось количество слов меньшее, либо равное чем установлено уровнем прерывания TLEV (MFBSP_TXBUF).

37.7 Работа MFBSP в режиме порта ввода-вывода общего назначения

Если многофункциональный порт выключен (LEN=0, SPI_I2S_EN =0), внешние лини LDAT[7:0], LCLK, LACK можно использовать как 10-разрядный двунаправленный порт ввода-вывода.

Если включен режим последовательного порта (SPI_I2S_EN = 1), незадействованные в организации последовательной передачи данных выводы LDAT[7:4] могут быть использованы в качестве вводов-выводов общего назначения. Единственным ограничением в данной ситуации является то, что для определения режима работы последовательного порта используются биты GPIO_DR[5:0], которые не должны меняться в процессе передачи данных по последовательному порту. Поэтому при управлении выводами общего назначения LDAT[7:4] (управляются битами DIR_MFBSP [9:6]) запись в регистр DIR_MFBSP необходимо проводить таким образом, чтобы текущие значения бит DIR_MFBSP [5:0] не менялись.

При работе в режиме выводов общего назначения данные с внешних выводов порта защелкиваются по положительному фронту тактового сигнала. Поэтому следует учитывать, что чтение данных с внешних выводов порта будет происходить с задержкой в 1 такт.

37.7.1 Регистр данных порта ввода вывода GPIO_DR

10-разрядный регистр данных порта ввода-вывода (GPIO_DR) предназначен для реализации гибкого интерфейса с внешними устройствами. Внешние выводы порта ввода-вывода совмещены с внешними выводами линкового порта.

Соответствие разрядов регистра GPIO_DR и внешних линий линкового порта приведено в Таблица 37.38.

Таблица 37.38. Назначение разрядов регистра GPIO_DR

Номер разряда Регистра GPIO_DR	Внешние выводы MFBSP	Значение после сброса
9:2	LDAT[7:0]	0
1	LCLK	0
0	LACK	0

37.7.2 Регистр управления направлением выводов DIR_MFBSP

Настройка направления выводов порта ввода-вывода осуществляется программно при помощи 10-разрядного регистра DIR_MFBSP. Если DIR_MFBSP установлен в 0, то соответствующий разряд порта ввода-вывода является входом, если же разряд DIR_MFBSP установлен в 1, то соответствующий разряд порта ввода-вывода является выходом.

Таблица 37.39. Назначение разрядов регистра DIR_MFBSP

Номер разряда Регистра DIR_MFBSP	Внешние выводы MFBSP	Значение после сброса
9:2	Направление выводов LDAT[7:0]	0
1	Направление вывода LCLK	0
0	Направление вывода LACK	0

37.8 Рекомендации по аварийному выключению передатчика

В режимах SPI и I2S при TDEL = 1 выключение порта путем записи 0 в TEN, без сброса бита SPI_I2S_EN может привести к сбою в буфере передачи, и после очередного включения передатчика (TEN=1) данные будут передаваться некорректно.

Решения:

- Если передатчик был выключен при TDEL=1 перед его очередным включением, необходимо сбросить записью 1 в бит RST_TXBUF
- В режиме мастер выключать передатчик (если есть необходимость в дальнейшем использовать порт) вообще нет необходимости – отсутствие данных в буфере передачи автоматически останавливает дальнейшую передачу.

37.9 MFBSP в режиме CAN. Общие сведения

37.9.1 Функциональные параметры и возможности

- совместим с CAN протоколом версии 2.0 A, B ISO 11898-1 и ARINC 825;
- поддержка режима планирования событий (time-triggering);
- 128 фильтров для стандартных идентификаторов;
- 64 фильтров для расширенных идентификаторов;
- буфер приёма Rx FIFO на 64 сообщения;
- буфер передачи Tx Buff на 32 сообщения;
- буфер Tx Event FIFO на 32 элемента подтверждения передачи;
- прямой доступ к RAM памяти и регистрам контроллера по CPU Data Bus;
- поддержка тестового режима LoopBack;
- поддержка маскируемых прерываний;
- отключение автоматического повтора передачи.

37.9.2 Совместимость протоколов CAN и ARINC825

Архитектура протоколов включает несколько уровней абстракции. Совместимость CAN протокола и ARINC825 протокола, накладывает требования на физический и канальный уровни протокола.

Совместимость на физическом уровне обеспечивается параметрами приёмопередатчиков.

Приёмопередатчики CAN должны быть полностью совместимы с ISO 11898-2 High Speed спецификацией (электрические параметры и их допуски – выходное напряжение, порог срабатывания). С включёнными средствами электромагнитной защиты, узел также должен обеспечивать полную совместимость с ISO 11898-2 High Speed спецификацией по скорости передачи данных. Допускаются скорости передачи по CAN шине до 1 Mbps.

Приёмопередатчик должен не допускать длительного удержания шины в доминантном состоянии. Защита должна отключать приёмопередатчик, когда доминанта удерживается дольше допустимого промежутка времени.

Узел не должен влиять на общую шину если на него не подано питание.

Совместимость на канальном уровне обеспечивается параметрами контроллера:

CAN контроллер должен быть полностью совместим со спецификацией на CAN2.0B (ISO 11898-1) и отвечать ISO 16845.

Наличие режима планирования событий (time-triggering) позволяет аппаратно контролировать пропускную способность CAN шины, при реализации протокола ARINC825.

37.9.3 Структурная схема

Контроллер включает в себя:

Память RAM – содержит буфера для исходящих сообщений Tx Buff, входящих сообщений Rx FIFO, буфер подтверждения передачи Tx Event FIFO, элементы событий и фильтры (см. п.37.10);

Модуль управление приёмом – сохраняет в буфер Rx FIFO сообщения прошедшие входную фильтрацию (см. п. 37.12.9).

Модуль входной фильтрации — принимает или отклоняет входящие сообщения в зависимости от конфигурации фильтра и идентификатора в заголовке сообщения.

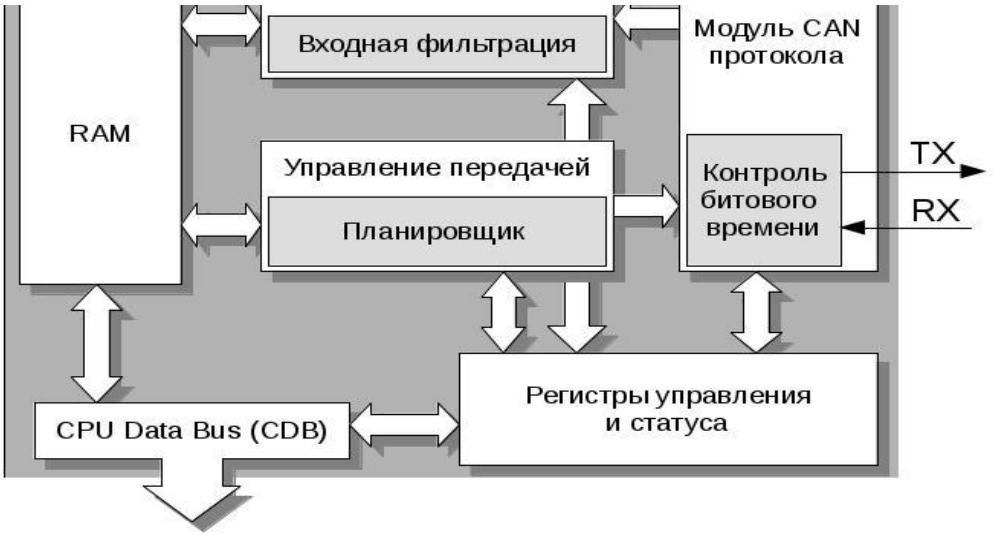
Модуль управление передачей – выбирает сообщение из внутренней памяти RAM, в зависимости от режима работы, и передаёт в модуль CAN протокола для последующей отправки (см. п37.12.10).

Планировщик — управляет передачей сообщений согласно конфигурации элементов событий (см. п. 37.12.12.8).

Модуль CAN протокола - обеспечивает выполнение CAN протокола (ISO 11898-1) и обмен данными с PHY.

Модуль контроля битового времени — обеспечивает скорость передачи согласно

Рисунок 37.40. CAN. Структурная схема



конфигурации регистра BTP.

Регистры управления и статуса – определяют режим работы контроллера и отражают статусную информацию.

CDB (CPU Data Bus) — интерфейс для доступа к регистрам и памяти RAM.

37.10 Адресное пространство контроллера CAN

Для хранения входящих и исходящих сообщении контроллер использует 32 разрядную память.

Таблица 37.40. Адресное пространство контроллера CAN

Адрес	Описание
0x0000-0x01FF	Регистры.
0x0200-0x03FF	Фильтры стандартных сообщений.
0x0400-0x05FF	Фильтры расширенных сообщений.
0x0600-0x0BFF	Элементы буфера Rx FIFO.
0x0C00-0x0EFF	Элементы буфера Tx Buff.
0x0F00-0x10FF	Элементы буфера Tx Event FIFO.
0x1100-0x12FF	Элементы событий.

37.10.1 Фильтры стандартных сообщений

Поддерживается 128 фильтров стандартных сообщений. Размер фильтра - одно 32-битное слово. Адрес фильтра вычисляется сложением начального адреса фильтров стандартных сообщений и индекса фильтра (0…127)x4. Описание формата фильтра стандартных сообщений приведено в следующей таблице:

Таблица 37.41. Формат фильтра стандартных сообщений.

Номер разряда	Условное обозначение	Назначение
31:30	SFT[1:0]	Способ фильтрации ID принимаемых сообщений: 00 - Диапазонный фильтр от SF1ID до SF2ID (SF2ID ≥ SF1ID); 01 - Двойной ID фильтр для SF1ID или/и SF2ID; 10 - Классический фильтр: SF1ID = фильтр, SF2ID = маска; 11 - Не используется.
29:27	SFEC[2:0]	Действие фильтра: 000 - элемент отключён; 001 - сохранить в буфере FIFO в случае соответствия; 010 - резерв; 011 - отклонить в случае соответствия; 100 - резерв; 101 - установить прерывание HPM и сохранить в случае соответствия; 110 - резерв; 111 - установить прерывание HPM и отклонить в случае соответствия;
26:16	SFID1[10:0]	Первый ID элемента фильтра.
15:11	-	Резерв
10:0	SFID2[10:0]	Второй ID элемента фильтра.

37.10.2 Фильтры расширенных сообщений

Поддерживается 64 фильтра расширенных сообщений. Размер фильтра в памяти - два 32-битных слова. Адрес фильтра вычисляется сложением начального адреса фильтров расширенных сообщений и индекса фильтра (0…63)x2x4. Описание формата фильтра расширенных сообщений приведено в следующей таблице:

Таблица 37.42. Формат фильтра расширенных сообщений

Номер разряда	Условное обозначение	Назначение
Слово 0		
31:29	EFEC[2:0]	Действие фильтра: 000 - элемент отключён; 001 - сохранить в буфере FIFO в случае соответствия; 010 - резерв; 011 - отклонить в случае соответствия; 100 - резерв; 101 - установить прерывание HPM и сохранить в случае соответствия; 110 - резерв; 111 - установить прерывание HPM и отклонить в случае соответствия;
28:0	EFID1[28:0]	Первый ID фильтра.
Слово 1		
31:30	EFT[1:0]	Способ фильтрации принимаемых кадров: 00 - Диапазонный фильтр от EF1ID до EF2ID (EF2ID ≥ EF1ID), XIDAM маска применяется; 01 - Двойной ID фильтр для EF1ID или/и EF2ID , XIDAM маска применяется; 10 - Классический фильтр: EF1ID = фильтр,ESF2ID = маска , XIDAM маска применяется; 11 - Диапазонный фильтр от EF1ID до EF2ID (EF2ID ≥ EF1ID), XIDAM маска не применяется.
29	-	Резерв
28:0	EFID2[28:0]	Второй ID фильтра.

37.10.3 Элемент буфера Rx FIFO

Rx FIFO содержит 64 элемента. Размер элемента - шесть 32-битных слов. Управление приёмным буфером, считывание его текущего состояния осуществляется через регистры RXF0C, RXF0S, RXF0A.

Сообщения, прошедшие фильтрацию, последовательно записываются в циклический буфер Rx FIFO. Чтение одного или нескольких элементов из буфера производится в удобном для ПО порядке, с произвольного элемента. Подтверждением того, что данные из Rx FIFO прочитаны, является запись в регистр RXF0A индекса элемента, который принимается контроллером за последний элемент считанный из Rx FIFO.

В следующей таблице приведено описание формата элемента буфера Rx FIFO:

Таблица 37.43. Формат элемента буфера RxFIFO

Номер разряда	Условное обозначение	Назначение
Слово 0		
31	-	Резерв
30	XTD	Тип идентификатора в принятом сообщении: 0 - 11-битный стандартный идентификатор; 1 - 29-битный расширенный идентификатор.
29	RTR	Состояние поля RTR (удалённого запроса) в принятом сообщении: 0 - принято сообщение с данными; 1- принят удалённый запрос.

28:0	ID	Значение поля идентификатор. Стандартный идентификатор записывается в ID[10:0].
Слово 1		
31	ANMF	Принято сообщение несоответствующее ни одному фильтру 0 - принятое сообщение содержит номер соответствующего фильтра в поле FIDX 1 - принятое сообщение не соответствует ни одному фильтру. Приём/отклонение сообщений не соответствующих ни одному из фильтров настраивается через GFC.ANFS и GFC.ANFE
30:24	FIDX	Номер соответствующего фильтра. В диапазоне от 0 до SIDFC.LSS - 1 (или XIDFC.LSE – 1).
23	MON_AE	MonitorAckError. Если, в режиме монитора (CCCR.MON=1) при отключённом режиме LoopBack (CCCR.LBM=0) отсутствовал сигнал подтверждения в линии разряд устанавливается в единицу.
22:21	-	Резерв
20	ASF	Принято собственное сообщение (контроллер успешно отправил и принял своё собственное сообщение). См. GFC.RSF
19:16	DLC	Размер поля данных принятого сообщения 0-8 байт.
15:0	RXTS	Значение Timestamp счётчика захваченное в начале приёма кадра.
Слово 2		
31:0	DBn[7:0]	Байты данных 3-0.
Слово 3		
31:0	DBn[7:0]	Байты данных 7-4.
Слово 4		
31:0	TS64[63:32]	Старшие разряды 64-х разрядного счётчика времени (значение регистра TS64H захваченное по SOF).
Слово 5		
31:0	TS64[31:0]	Младшие разряды 64-х разрядного счётчика времени (значение регистра TS64L захваченное по SOF).

37.10.4 Элемент буфера Tx Event FIFO.

Буфер Tx Event FIFO содержит 32 элемента. Размер элемента в памяти - четыре 32-битных слова. Адрес элемента буфера вычисляется сложением начального адреса буфера Tx Event FIFO и индекса элемента (0…31) x 2.

Таблица 37.44. Формат элемнта буфера TX Event FIFO

Номер разряда	Условное обозначение	Назначение
Слово 0		
31	-	Резерв
30	XTD	Тип идентификатора в переданном сообщения: 0 - 11-битный стандартный идентификатор; 1 - 29-битный расширенный идентификатор.
29	RTR	Состояние поля RTR (удалённого запроса) в переданном сообщении: 0 - передан кадр данных; 1 - передан кадр удалённого запроса.
28:0	ID[28:0]	Значение поля идентификатор. Стандартный идентификатор записывается в ID[10:0].
Слово 1		
31:24	MM[7:0]	Маркер сообщения. Записывается в процессе конфигурации буфера. Копируется в сообщение Tx Event FIFO для идентификации переданного сообщения.
23:20	-	Резерв
19:16	DLC	Размер поля данных передаваемого сообщения 0-8 байт.

15:0	TXTS[15:0]	Значение Timestamp счётчика захватывается в начале приёма кадра. Разрешение зависит от конфигурации делителя Timestamp Counter Prescaler TSCC.TCP.
Слово 2		
31:0	TS64[63:32]	Старшие разряды 64-х разрядного счётчика времени (значение регистра TS64H захваченное по SOF).
Слово 3		
31:0	TS64[31:0]	Младшие разряды 64-х разрядного счётчика времени (значение регистра TS64H захваченное по SOF).

37.10.5 Элемент буфера Tx Buff

Буфер Tx Buff содержит 32 элемента. Размер элемента в памяти - четыре 32-битных слова. Адрес элемента буфера вычисляется сложением начального адреса буфера Tx Buff и индекса элемента (0…31) x 4.

Таблица 37.45. Формат элемента буфера TX Buff

Номер разряда	Условное обозначение	Назначение
Слово 0		
31	-	Резерв
30	XTD	Тип идентификатора в передаваемом сообщении: 0 - 11-битный стандартный идентификатор; 1 - 29-битный расширенный идентификатор.
29	RTR	Состояние поля RTR (удалённого запроса) в передаваемом сообщении: 0 - кадр данных; 1 - кадр удалённого запроса.
28:0	ID	Идентификатор. Стандартный идентификатор записывается в ID[10:0].
Слово 1		
31:24	MM[7:0]	Маркер сообщения. Записывается в процессе конфигурации буфера. Копируется в сообщение Tx Event FIFO для идентификации переданного сообщения.
23	EFC	Разрешение для Tx Event FIFO: 0 – не сохранять данные о передаче; 1 – сохранять данные о передаче.
22:20	-	Резерв
19:16	DLC	Размер поля данных передаваемого сообщения 0-8 байт.
15:6	-	Резерв
5:0	PRI	Поле приоритета сообщения. Наиболее приоритетным считается сообщение с наименьшим значением поля PRI. Значение PRI используется при разряде TXBC.PRI_MODE установленном в единицу, иначе наиболее приоритетным считается сообщение с наименьшим ID.
Слово 2		
31:0	DBn[7:0]	Байты данных 3-0
Слово 3		
31:0	DBn[7:0]	Байты данных 7-4
Слово 4		
31:0	TO[63:32]	Старшие разряды значения 64-х разрядного счётчика времени, после которого выдача сообщения автоматически отменяется
Слово 5		
31:0	TO[31:0]	Младшие разряды значения 64-х разрядного счётчика времени, после которого выдача сообщения автоматически отменяется

37.10.6 Элемент события

До 64 элементов события может использоваться. Размер элемента в памяти - два 32-битных слова. Адрес элемента вычисляется сложением начального адреса элементов событий и индекса элемента (0…63) x 2. Изменение элемента возможно только при CCCR.INIT=1.

Таблица 37.46. Формат элемента события

Номер разряда	Условное обозначение	Назначение
Слово 0		
31:16	TM[15:0]	Метка времени элемента события. Определяет момент в который событие будет выполнено.
15	-	Резерв
14:8	CC	Номер цикла, в котором выполнится событие: (кроме событий Tx_Ref_Trigger(_Gap), Watch_Trigger(_Gap), End_of_List) 0b000000x активно каждый цикл; 0b000001c активно каждый 2 цикл начиная с номера c; 0b00001cc активно каждый 4 цикл начиная с номера cc; 0b0001ccc активно каждый 8 цикл начиная с номера ccc; 0b001cccc активно каждый 16 цикл начиная с номера cccc; 0b01ccccc активно каждый 32 цикл начиная с номера ccccc; 0b1cccccc активно каждый 64 цикл начиная с номера cccccc.
7:4	-	Не используются.
3:0	TYPE[3:0]	Тип события: 0000 – Tx_Ref_Trigger; 0001 – Tx_Ref_Trigger_Gap; 0010 – Tx_Trigger_Single; 0011 – Tx_Trigger_Continous; 0100 – не используется; 0101 – не используется; 0110 – Watch_Trigger; 0111 – Watch_Trigger_Gap; 1000…1111 – End_of_List.
Слово 1		
31:23	-	Резерв
22:16	MNR[6:0]	Содержит номер передаваемого буфера 0..31.
15:0	-	Резерв

37.11 Регистры контроллера CAN

Регистры контроллера доступны по шине CDB. Запись в регистры/биты, отмеченные как «P – защищённая запись» возможна только при CCCR.CCE = “1” и CCCR.INIT=”1”. При записи в поле, отмеченное как «S – установка по чтению», соответствующие разряды установятся в «1» после чтения. При записи в поле, отмеченное как «X – сброс по чтению», соответствующие разряды установятся в ”0” после чтения. Возможна задержка в несколько тактов между записью значения в командный регистр и отображением результата в статусном регистре.

Таблица 37.47. Регистры контроллера CAN

Смещение	Условное обозначение	Название	Значение по сбросу	Доступ
0x000-0x014	-	Резерв	0000 0000	R

Смещение	Условное обозначение	Название	Значение по сбросу	Доступ
0x018	CCCR	Регистр контроля и конфигурации	0000 0001	RWP
0x01C	BTP	Регистр конфигурации скорости передачи	0000 0A33	RP
0x020	TSCC	Регистр конфигурации счётчика Timestamp	0000 0000	RP
0x024	TSCV	Регистр значения счётчика Timestamp	0000 0000	RC
0x028-02C	-	Резерв	0000 0000	R
0x030	TS64CTR	Регистр управления 64-разрядной метки времени		RW
0x034	TS64H	Старшие разряды 64-разрядной метки времени		RW
0x038	TS64L	Младшие разряды 64-разрядной метки времени		RW
0x03C	-	Резерв	0000 0000	R
0x040	ECR	Регистр счётчика ошибок CAN	0000 0000	RW
0x044	PSR	Регистр статуса протокола CAN	0000 0007	RXS
0x048-04C	-	Резерв	0000 0000	R
0x050	IR	Регистр прерываний	0000 0000	RW
0x054	IE	Регистр разрешения прерывания	0000 0000	RW
0x058-07C	-	Резерв	0000 0000	R
0x080	GFC	Регистр общей конфигурации фильтров	0000 0000	RP
0x084	SIDFC	Регистр конфигурации фильтров стандартных сообщений	0000 0000	RP
0x088	XIDFC	Регистр конфигурации фильтров расширенных сообщений	0000 0000	RP
0x08C	-	Резерв	0000 0000	R
0x090	XIDAM	Регистр маски для фильтров расширенных сообщений	1FFF FFFF	RP
0x094	HPMS|-	Регистр состояния приоритетного сообщения|Не используется	0000 0000	R
0x098-0x09C	-	Резерв	0000 0000	R
0x0A0	RXF0C	Регистр конфигурации буфера Rx FIFO	0040 0000	RP
0x0A4	RXF0S	Регистр статуса буфера Rx FIFO	0000 0000	R
0x0A8	RXF0A	Регистр подтверждения считывания буфера Rx FIFO	0000 0000	R
0x0AC-0x0BC	-	Резерв	0000 0000	R
0x0C0	TXBC	Регистр конфигурации буфера Tx Buffer	0000 0000	R
0x0C4-0x0C8	-	Резерв	0000 0000	R
0x0CC	TXBRP	Регистр занятости буфера передачи Tx Buffer	0000 0000	R
0x0D0	TXBAR	Регистр запроса передачи буфера Tx Buffer	0000 0000	RW
0x0D4	TXBCR	Регистр запроса отмены передачи буфера Tx Buffer	0000 0000	RW
0x0D8	TXBTO	Регистр окончания передачи буфера Tx Buffer	0000 0000	R
0x0DC	TXBCF	Регистр окончания отмены передачи буфера Tx Buffer	0000 0000	R
0x0E0	TXBTIE	Регистр разрешения прерывания по окончании передачи буфера Tx Buffer	0000 0000	RW
0x0E4	TXBCIE	Регистр разрешения прерывания по окончании отмены передачи буфера Tx Buffer	0000 0000	RW
0x0E8-0EC	-	Резерв	0000 0000	R
0x0F0	TXEFC	Регистр конфигурации буфера Tx Event FIFO.	0000 0000	RP
0x0F4	TXEFS	Регистр статуса буфера Tx Event FIFO.	0000 0000	R
0x0F8 - 0x100	-	Резерв	0000 0000	R
0x104	TTRMC	Регистр конфигурации опорного сообщения.	0000 0000	RP
0x108	TTOCF	Регистр конфигурации режима планирования событий.	0001 0000	RP
0x10C	TTMLM	Регистр конфигурации системной матрицы.	0000 0000	RP
0x110	TURCF	Регистр конфигурации делителя TUR.	1000 0000	RP
0x114	TTOCN	Регистр управления режимом планирования событий.	0000 0000	RW
0x118 - 0x11C	-	Резерв	0000 0000	R
0x120	TTIR	Регистр прерываний в режиме планирования событий.	0000 0000	RW

Смещение	Условное обозначение	Название	Значение по сбросу	Доступ
0x124	TTIE	Регистр разрешения прерываний в режиме планирования событий.	0000 0000	RW
0x128	-	Резерв	0000 0000	R
0x12C	TTOST	Регистр статуса в режиме планирования событий.	0000 0080	R
0x130	TURNA	Регистр текущего значения делителя TUR	0001 0000	R
0x134	TTLGT	Регистр локального и глобального времени	0000 0000	R
0x138	TTCTC	Регистр времени цикла и номера цикла	0000 0000	R
0x13C- 0x14C	-	Резерв	0000 0000	R
0x150	STATTX	Число ошибок при передаче	00000000	RW
0x154	STATRX	Число ошибок при приёме	00000000	RW
0x158	STATWRN	Число превышений уровня предупреждения	00000000	RW
0x15C	STATEP	Число переходов в состояние Error Passive	00000000	RW
0x160	STATBO	Число переходов в состояние Bus_Off	00000000	RW
0x164	STATLOST	Число потерянных сообщений при приёме	00000000	RW

37.11.1 CCCR - Регистр контроля и конфигурации

Формат регистра CCCR приведён в Таблица 37.48.

Таблица 37.48. Формат регистра CCCR

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:7	-	Резерв	R	0x0
6	DAR	отключение повторной передачи: 0 – автоматический повтор передачи включён; 1 – автоматический повтор передачи отключён.	RP	0x0
5	MON	Управление режимом монитора: 0 - режим монитора отключён; 1 - режим монитора включён.	RP	0x0
4	CSR	Запрос на отключение тактов: 0 – нет запроса на отключение тактов; 1 – запрос на отключение тактов.	RW	0x0
3	CSA	Готовность к отключению тактов: 0 – нет подтверждения готовности отключения тактов; 1 – такты можно отключить.	R	0x0
2	LBM	Управление режимом Loop back: 0 – нормальный режим работы; 1 – Loop Back режим включен.	RP	0x0
1	CCE	Управление защищённым режимом: 0 – доступ к защищённым регистрам разрешён; 1 – доступ к защищённым регистрам разрешён при CCCR.INIT = “1”.	RP	0x0
0	INIT	Управление режимом инициализации: 0 – нормальный режим; 1 – режим инициализации.	RW	0x1

37.11.2 BTP - Регистр конфигурации скорости передачи

Регистр доступен для записи только при CCCR.CCE=“1” и CCCR.INIT=“1”. Длительность CAN бита программно задаётся от 4 до 81 tq (квант). Длительность кванта также программируемая- от 1 до 1024 периодов HCLK. tq = (BRP + 1) периодов HCLK . Поле TSEG1 это объединенная длительность сегментов Prop_Seg и Phase_Seg1. Поле TSEG2

задает длятельность Phase_Seg2. Таким образом длительность бита это [TSEG1 + TSEG2 + 3] tq или [Sync_Seg + Prop_Seg + Phase_Seg1 + Phase_Seg2] tq.

Например, для HCLK = 200 МГц скорости передачи:

1000 Кбит – BRP = 19, TSEG1 = 4, TSEG2 = 3;

500 Кбит – BRP = 19, TSEG1 = 10, TSEG2 = 7;

250 Кбит – BRP = 19, TSEG1 = 20, TSEG2 = 17;

125 Кбит – BRP = 19, TSEG1 = 40, TSEG2 = 37;

Формат регистра BTP приведён в следующей таблице:

Таблица 37.49. Формат регистра BTP

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:26	-	Резерв	R	0x0
25:16	BRP[9:0]	Делитель тактовой частоты	RP	0x0
15:14	-	Резерв	R	0x0
13:8	TSEG1[5:0]	Длительность сегмента до семплирования	RP	0xA
7:4	TSEG2[3:0]	Длительность сегмента после семплирования	RP	0x3
3:0	SJW[3:0]	Длительность сегмента ресинхронизации	RP	0x3

37.11.3 TSCC - Регистр конфигурации счётчика Timestamp

Назначение регистра описано в Разделе 37.12.7. Формат регистра TSCC приведён в следующей таблице:

Таблица 37.50. Формат регистра TSCC

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:20	-	Резерв	R	0x0
19:16	TCP[3:0]	Делитель счётчика Timestamp	RP	0x0
15:2	-	Резерв	R	0x0
1:0	TSS[1:0]	00 – значение Timestamp счётчика всегда 0x0000 01 – значение Timestamp счётчика инкрементируется в соответствии с TCP 10 – эквивалентно “00” 11 – эквивалентно “00”	RP	0x0

37.11.4 TSCV - Регистр значения счётчика Timestamp

Формат регистра TSCV приведён в следующей таблице:

Таблица 37.51. Формат регистра TSCV

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:16	-	Резерв	R	0x0

15:0	TSC[15:0]	Значение счётчика Timestamp.	RC	0x0

37.11.5 TS64CTR - Регистр управления 64-х разрядной метки времени

Формат регистра TS64CTR приведён в следующей таблице:

Таблица 37.52. Формат регистра TS64CTR

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31	en_cancel_by_ts64	Разрешение отмены пересылки сообщения по значению 64-х разрядной метки времени. 0. Отмена отключена. 1. Отмена разрешена. Если при выдаче сообщения контроллер определяет, что поле Tx Buff.TO меньше текущего значения TS64, передача отменяется. Передачи также отменяется в моменты выбора нового сообщения.	RW	0x0
30:24	-		R	0x0
23:4	div_max	Длительность периода срабатывания счётчика метки времени. При достижении внутренним делителем частоты значения div_max на выходе TSO_incr формируется импульс длительностью 3 такта системной частоты. Внимание! При использовании сигнала TSO_incr значение div_max должно быть > 6 (для уверенного выделения фронта и спада сигнала TSO_incr другим устройством).	RW	0x0
3	use_ext_incr	Условие срабатывания счётчика метки времени 0 Срабатывание по внутреннему делителю частоты (см. поле div_max) 1 Срабатывание по фронту входного сигнал TSI_incr	RW	0x0
2	use_ext_reset	Условие сброса счётчика метки времени 0 Разрешён только программный сброс 1 Разрешен сброс по фронту входного сигнала can_TSI_reset	RW	0x0
1	reset	Сброс метки времени. После записи единицы разряд сбрасывается автоматически. На выходе can_TSO_reset формируется импульс длительностью 3 такта системной частоты.	RW	0x0
0	enable	Разрешение работы счётчика метки времени	RW	0x0

37.11.6 TS64H - Старшие разряды 64-х разрядной метки времени

Формат регистра TS64H приведён в следующей таблице:

Таблица 37.53. Формат регистра TS64H

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:0	TS64[63:32]	Старшие разряды 64-х разрядной метки времени. После достижения максимального значения 0xFFFFFFFF_FFFFFFFF увеличение метки времени прекращается, метка сохраняет своё значение. Для продолжения подсчёта необходимо записать в регистр требуемое начальное значение, счёт продолжится с	RW	0x0

		этого значения(или же произвести сброс см. TS64CTR.reset).		

37.11.7 TS64L - Младшие разряды 64-х разрядной метки времени

Формат регистра TS64L приведён в следующей таблице:

Таблица 37.54. Формат регистра TS64L

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:0	TS64[31:0]	Младшие разряды 64-х разрядной метки времени. После достижения максимального значения 0xFFFFFFFF_FFFFFFFF увеличение метки времени прекращается, метка сохраняет своё значение. Для продолжения подсчёта необходимо записать в регистр требуемое начальное значение, счёт продолжится с этого значения (или же произвести сброс см. TS64CTR.reset)	RW	0x0

37.11.8 ECR - Регистр счётчика ошибок CAN

Счётчики ошибок приёма REC и передачи TEC функционируют согласно спецификации CAN 2.0 B. Формат регистра ECR приведён в следующей таблице:

Таблица 37.55. Формат регистра ECR

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:16	-	Резерв	R	0x0
15:8	REC[7:0]	Счётчик ошибок приёма, функционирует согласно спецификации CAN 2.0B. При достижении значения большего 127 инкрементация счётчика прекращается. При входе в режим Bus_Off значение счётчика сбрасывается в ноль. Разряды REC[6:0] доступны по записи для тестовых целей (старший разряд при записи обнуляется).	RW	0x0
7:0	TEC[7:0]	Счётчик ошибок передачи, функционирует согласно спецификации CAN 2.0B. При достижении значения более 256 (контроллер переходит в состояние Bus_Off) TEC принимает значение 128 и начинает декрементироваться на единицу при каждом обнаружении 11 непрерывно идущих рецессивных битов в шине. Счётчик TEC доступен по записи для тестовых целей.	RW	0x0

37.11.9 PSR - Регистр статуса протокола CAN

Формат регистра PSR приведён в следующей таблице:

Таблица 37.56. Формат регистра PSR

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:8	-	Резерв	R	0x0

7	BO	Режим Bus Off. Режим активен если значение TEC более 255 (см. спецификацию CAN 2.0B). 0 – контроллер не в состоянии Bus_Off; 1 – контроллер в состоянии Bus_Off.	R	0x0
6	EW	Флаг Error Warning (см. спецификацию CAN 2.0B): 0 – счётчики ошибок не достигли значения 96 1 – один из счётчиков достиг значения 96 При входе в режиме Bus_Off признак сбрасывается в 0.	R	0x0
5	EP	Режим Error Passive. Режим активен если значение одного из счётчиков ошибок (TEC или REC) более 127 (см. спецификацию CAN 2.0B). 0 – контроллер находится в состоянии Error_Active. Он участвует в операциях на шине и посылает error кадр если обнаруживает ошибку; 1 – контроллер находится в состоянии Error_Passive. При входе в режиме Bus_Off признак сбрасывается в 0.	R	0x0
4:3	-	Резерв	R	0x0
2:0	LEC[2:0]	Код последней ошибки на шине (см. спецификацию CAN 2.0B): 0 - ошибок нет (устанавливается после каждого успешного приёма или передачи); 1 - Stuff Error: принято более 5 одинаковых бит последовательно; 2 - Form Error: принятое сообщение имеет не верный формат; Принят кадр в котором хотя бы в одном из полей «Разделитель CRC (CRC delimiter)», «Разделитель ACK (ACK delimiter)», «Конец кадра (EOF)» был детектирован доминантый уровень. 3 - AckError: сообщение переданное контроллером не было подтверждено (acknowledged) другим узлом; 4 - Bit1Error: во время передачи сообщения (исключая поля арбитрации); контроллер пытался передать рецессивный уровень (лог «1»), а принял доминантный; 5 - Bit0Error: во время передачи сообщения (или бита acknowledge , или флага active error, или флага overload), пытался передать доминантный уровень (лог «0»), но принял рецессивный; 6 - CRCError: ошибка контрольной суммы; 7 - без изменений: при чтении регистра LEC = “7”. Если были одновременно детектированы несколько ошибок, приоритет имеет ошибка с наименьшим кодом.	S	0x7

37.11.10 IR - Регистр прерываний

Флаг прерывания устанавливается по фронту сигнала соответствующего события.

Флаг остаётся установленным пока не будет сброшен. Сброс происходит при записи «1» в соответствующий разряд. Флаги коммутируются по логическому «или» на INT0. Аппаратный сброс очищает регистр. Разрешение прерывания конфигурируется в регистре IE. Формат регистра IR приведён в следующей таблице:

Таблица 37.57. Формат регистра IR

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31	STE	Ошибка стаффинга (Stuff Error). Ошибка определяется, если при приёме кадра декодируется 6 последовательных битов с одинаковым значением, см. ISO 11898-1	RW	0x0
30	FOE	Ошибка формата (Format Error). Ошибка определяется, если биты имеющие фиксированный формат содержат одно или несколько недопустимых значений, см. ISO 11898-1	RW	0x0
29	ACKE	Ошибка подтверждения (Acknowledge Error). Ошибка определяется, если передатчик при выдаче кадра не получает сигнал подтверждения в интервале ACK, см. ISO 11898-1	RW	0x0
28	BE	Ошибка бита (Bit Error). Ошибка определяется, если при контроле битов выдаваемых в шину, передатчик обнаруживает в шине бит отличный от переданного (за исключением приёма доминантного бита при арбитраже и др., см. ISO 11898-1)	RW	0x0
27	CRCE	Ошибка контрольной суммы (CRC Error).	RW	0x0
26	-	Резерв	RW	0x0
25	BO	Переход в состояние Bus_Off.	RW	0x0
24	EW	Изменение статуса Error_Warning.	RW	0x0
23	EP	Переход в состояние Error_Passive.	RW	0x0
22:17	-	Резерв	R	0x0
16	TSW	Переполнение счётчика Timestamp.	RW	0x0
15	TEFL	Потеря элемента Tx Event FIFO. Сообщение успешно завершено, но буфер Tx Event FIFO заполнен, элемент события не сохраняется.	RW	0x0
14	TEFF	Буфер Tx Event FIFO заполнен.	RW	0x0
13	TEFW	Количество сообщений в буфере Tx Event FIFO достигло TXEFC.EFWM.	RW	0x0
12	TEFN	Новое сообщение в буфере Tx Event FIFO.	RW	0x0
11	-	Резерв	R	0x0
10	TCF	Завершена отмена передачи сообщения.	RW	0x0
9	TC	Завершена передача сообщения.	RW	0x0
8	HPM	Принято высокоприоритетное сообщение.	RW	0x0
7:4	-	Резерв	R	0x0
3	RF0L	Произошла запись в буфер Rx FIFO после его заполнения.	RW	0x0
2	RF0F	Буфер Rx FIFO заполнен целиком.	RW	0x0
1	RF0W	Количество сообщений в буфере Rx FIFO достигло RXF0C.F0WM	RW	0x0
0	RF0N	Новое сообщение в буфере Rx FIFO.	RW	0x0

37.11.11 IE - Регистр разрешения прерываний

Регистр разрешения прерываний. 0 – прерывание запрещено, 1 – прерывание разрешено. Формат регистра IE приведён в следующей таблице:

Таблица 37.58. Формат регистра IE

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31	STEE	Ошибка стаффинга (Stuff Error).	RW	0x0

30	FOEE	Ошибка формата (Format Error).	RW	0x0
29	ACKEE	Ошибка подтверждения (Acknowledge Error).	RW	0x0
28	BEE	Ошибка бита (Bit Error).	RW	0x0
27	CRCEE	Ошибка контрольной суммы (CRC Error).	RW	0x0
26	-	Резерв	R	0x0
25	BOE	Переход в состояние Bus_Off.	RW	0x0
24	EWE	Изменение статуса Error_Warning.	RW	0x0
23	EPE	Переход в состояние Error_Passive.	RW	0x0
22:17	-	Резерв	R	0x0
16	TSWE	Переполнение счётчика Timestamp.	RW	0x0
15	TEFLE	Разрешение прерывание IR.TEFL	RW	0x0
14	TEFFE	Буфер Tx Event FIFO заполнен.	RW	0x0
13	TEFWE	Количество сообщений в буфере Tx Event FIFO достигло TXEFC.EFWM.	RW	0x0
12	TEFNE	Новое сообщение в буфере Tx Event FIFO.	RW	0x0
11	-	Резерв	R	0x0
10	TCFE	Завершена отмена передачи сообщения.	RW	0x0
9	TCE	Завершена передача сообщения.	RW	0x0
8	HPME	Разрешение прерывания HPM «Принято высокоприоритетное сообщение».	RW	0x0
7:4	-	Резерв	R	0x0
3	RF0LE	Произошла запись в буфер Rx FIFO после его заполнения.	RW	0x0
2	RF0FE	Буфер Rx FIFO не пустой.	RW	0x0
1	RF0WE	Количество сообщений в буфере Rx FIFO достигло RXF0C.F0WM	RW	0x0
0	RF0NE	Новое сообщение в буфере Rx FIFO.	RW	0x0

37.11.12 GFC - Регистр общей конфигурации фильтров

Регистр GFC содержит глобальные настройки для фильтрации сообщений. Формат регистра GFC приведён в следующей таблице:

Таблица 37.59. Формат регистра GFC

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:7	-	Резерв	R	0x0
6	RSF	Запрет приёма собственных сообщений 0 - Приём собственных сообщений разрешён 1 - Приём собственных запрещён	RP	0x0
5:4	ANFS[1:0]	Определяет поведение контроллера при приёме сообщений с 11-битным идентификатором, для которых не сработал ни один из фильтров: 00 – Сообщение сохраняется в Rx FIFO; 01 – Сообщение сохраняется в Rx FIFO; 10 – Сообщение отклоняется; 11 – Сообщение отклоняется;	RP	0x0
3:2	ANFE[1:0]	Определяет поведение контроллера при приёме сообщений с 29-битным идентификатором, для которых не сработал ни один из фильтров: 00 – Сообщение сохраняется в Rx FIFO; 01 – Сообщение сохраняется в Rx FIFO; 10 – Сообщение отклоняется; 11 – Сообщение отклоняется;	RP	0x0

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
1	RRFS	Запрет стандартных удалённых запросов*: 0 – принимать стандартные удалённые запросы; 1 – отклонять стандартные удалённые запросы;	RP	
0	RRFE	Запрет расширенных удалённых запросов: 0 – принимать расширенные удалённые запросы; 1 – отклонять расширенные удалённые запросы.	RP	

*удаленным запросом считается сообщение, в заголовке которого поле RTR=1.

37.11.13 SIDFC - Регистр конфигурации фильтров стандартных сообщений

Настройки для фильтрации сообщений со стандартным идентификатором. Формат регистра SIDFC приведён в следующей таблице:

Таблица 37.60. Формат регистра SIDFC

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:24	-	Резерв	R	0x0
23:16	LSS[7:0]	Количество фильтров стандартных сообщений: 0 – фильтры стандартных кадров отключены; 1-128 – количество фильтров; >128 – количество фильтров = 128;	RP	0x0
15:0	-	Резерв	R	0x0

37.11.14 XIDFC - Регистр конфигурации фильтров расширенных сообщений

Настройки для фильтрации сообщений с расширенным идентификатором. Формат регистра XIDFC приведён в следующей таблице:

Таблица 37.61. Формат регистра XIDFC

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:23	-	Резерв	R	0x0
22:16	LSE[6:0]	Количество фильтров стандартных сообщений: 0 – фильтры расширенных кадров отключены; 1-64 – количество фильтров; >64 – количество фильтров = 64;	RP	0x0
15:0	-	Резерв	R	0x0

37.11.15 XIDAM - Регистр маски для фильтров расширенных сообщений

При входной фильтрации, для сообщений с расширенным идентификатором, помимо самого фильтра применяется дополнительная маска. Маска накладывается по логическому “и”. Формат регистра XIDAM приведён в следующей таблице:

Таблица 37.62. Формат регистра XIDAM

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:29	-	Резерв	R	0x0
28:0	EIDM	Extended ID Mask.	RP	0x1FFFFFFF

37.11.16 HPMS - Регистр состояния приоритетного сообщения

Регистр сбрасывается при установленном CCCR.CCE.

Регистр обновляется после приёма сообщения для которого сработал фильтр с установленным признаком высокого приоритета. Формат регистра HPMS приведён в следующей таблице:

Таблица 37.63. Формат регистра HPMS

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:16	-	Резерв	R	0x0
15	FLST	Filter List. 0 — сработал фильтр стандартных сообщений 1 — сработал фильтр расширенных сообщений	R	0x0
14:8	FIDX[6:0]	Filter Index. Индекс сработавшего фильтра. Принимает значения от 0 до SIDFC.LSS - 1(XIDFC.LSE - 1)	R	0x0
7:6	MSI[1:0]	Message Storage Indicator. Признак сохранения соощения. 00 - Сообщение не сохранялось в Rx FIFO 01 - Сообщение потеряно из-за занятости Rx FIFO 10,11 - Сообщение сохранено в Rx FIFO	R	0x0
5:0	BIDX[5:0]	Buffer Index. Номер элемента Rx FIFO, куда было сохранено сообщение. Поле используется только при MSI[1] =1.	R	0x0

37.11.17 RXF0C - Регистр конфигурации буфера Rx FIFO

Формат регистра RXF0C приведён в следующей таблице:

Таблица 37.64. Формат регистра RXF0C

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31	F0OM	Режим работы Rx FIFO. 0 — Rx FIFO в блокирующем режиме. При полном заполнении Rx FIFO сохранение новых элементов блокируется. 1 — FIFO в режиме перезаписи При полном заполнении Rx FIFO самый старый элемент будет перезаписан новым.	RP	0x0
30:24	F0WM	Значение Watermark определяет количество данных в буфере Rx FIFO при котором сформируется прерывание IR.RF0W: 0 - Watermark прерывание отключено; 1-64 значение watermark Rx FIFO ; >64 Watermark прерывание отключено.	RP	0x0
23	-	Резерв	R	0x0

22:16	F0S	Rx FIFO Size. Размер Rx FIFO — количество элементов. 0 - Rx FIFO отсутствует 1-64 - Количество элементов Rx FIFO >64 - Интерпретируется как 64	R	0x40
15:2	-	Резерв	R	0x0
1:0	-	Резерв	R	0x0

37.11.18 RXF0S - Регистр статуса буфера Rx FIFO

Формат регистра RXF0S приведён в следующей таблице:

Таблица 37.65. Формат регистра RXF0S

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:26	-	Резерв	R	0x0
25	RF0L	Признак потери нового сообщения 0 - Потери сообщений нет. 1 - Rx FIFO заполнено, пришло новое сообщение, новое сообщение не будет записано в память. Признак совпадает с сигналом IR.RF0L.	R	0x0
24	F0F	Признак заполненного буфера Rx FIFO: 0 - буфер не полон; 1 - буфер полон.	R	0x0
23:22	-	Резерв	R	0x0
21:16	FOPI	Указатель записи RxFIFO. Индекс элемента в который будет произведена запись нового сообщения.	R	0x0
15:14	-	Резерв	R	0x0
13:8	FOGI	Указатель чтения RxFIFO. Индекс элемента из которого нужно начинать чтение Rx FIFO.	R	0x0
7	-	Резерв	R	0x0
6:0	F0FL	Количество сообщений в буфере Rx FIFO (0 - 64).	R	0x0

37.11.19 RXF0A - Регистр подтверждения считывания буфера Rx FIFO

Формат регистра RXF0A приведён в следующей таблице:

Таблица 37.66. Формат регистра RXF0A

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:26	-	Резерв	R	0x0
5:0	F1AI	Индекс последнего элемента считанного из Rx FIFO.	RW	0x0

37.11.20 TXBC - Регистр конфигурации буфера передачи Tx Buff

Формат регистра TXBC приведён в следующей таблице:

Таблица 37.67. Формат регистра TXBC

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:1	-	Резерв	R	0x0
0	PRI_MODE	Способ определения наиболее приоритетного сообщения при передаче.	RW	0x0

		0 — Наиболее приоритетным считается сообщение с наименьшим CAN ID. 1 — Наиболее приоритетным считается сообщение с наименьшим значением поля PRI элемента Tx Buff		

37.11.21 TXBRP - Регистр занятости буфера передачи Tx Buffer

Отражает состояние занятости буфера передачи Tx Buff. Биты устанавливаются в регистре TXBAR. Сбрасываются после окончания запрашиваемой передачи или её отмены с помощью регистра TXBCR, или же отмены по таймауту (см TS64*).

После установки регистра TXBRP, буфера сканируются, выбирается буфер с наивысшим приоритетом запроса на передачу (см. TXBC.PRI_MODE).

Запрос на отмену передачи сбрасывает соответствующий бит запроса на передачу в регистре TXBRP. Если передача уже началась к моменту возникновения запроса на отмену, отмена передачи завершится после окончания передачи, не важно была ли передача успешной или нет. Бит запроса отмены передачи сбросится сразу после сброса соответствующего бита TXBRP.

Завершение отмены отражается в регистре TXBCF в случаях:

- после успешной передачи вместе с соответствующим битом TXBTO;
- когда передача ещё не началась на момент отмены;
- когда передача была прервана из-за проигрыша арбитража;
- когда произошла ошибка во время передачи кадра;
- когда произошла отмена пересылки по таймауту TS64.

В режиме отмены повторной передачи (DAR) все передачи отменяются автоматически если они не закончились успешно. Соответствующий бит TXBCF устанавливается для всех неудачных передач.

Формат регистра TXBRP приведён в следующей таблице:

Таблица 37.68. Формат регистра TXBRP

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:0	TRP31-TRP0	Биты устанавливаются в “1” при запросе на передачу для соответствующего элемента Tx Buffer.	R	0x0

37.11.22 TXBAR - Регистр запроса передачи для буфера Tx Buffer

Устанавливает запрос на передачу для буфера Tx Buff. Допускается запись в несколько разрядов одновременно. Если буфер уже занят (установлен TXBRP), то запись в соответствующий бит игнорируется. Если в момент записи, процесс сканирования буферов

не запущен, то записываемый бит сбрасывается немедленно, в противном случае остаётся установленным до окончания сканирования.

Формат регистра TXBAR приведён в следующей таблице:

Таблица 37.69. Формат регистра TXBAR

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:0	AR31-AR0	Запись “1” в соответствующий разряд активирует запрос на передачу соответствующего элемента Tx Buffer в регистре TXBRP.	RW	0x0

37.11.23 TXBCR - Регистр запроса отмены передачи буфера Tx Buffer

Устанавливает запрос на отмену передачи для буфера Tx Buff. Допускается запись в несколько разрядов одновременно. Бит остаётся установленным пока установлен соответствующий бит TXBRP.

Формат регистра TXBCR приведён в следующей таблице:

Таблица 37.70. Формат регистра TXBCR

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:0	CR31-CR0	Запись “1” в соответствующий разряд отменяет запрос на передачу соответствующего элемента Tx Buffer.	RW	0x0

37.11.24 TXBTO - Регистр подтверждения передачи буфера Tx Buffer

Отражает состояние успешного завершения передачи для буфера Tx Buff. Бит TXBTO устанавливается, когда сбрасывается соответствующий бит TXBRP в результате успешной передачи. Бит TXBTO сбрасывается, когда происходит запись «1» в соответствующий разряд TXBAR. Формат регистра TXBTO приведён в следующей таблице:

Таблица 37.71. Формат регистра TXBTO

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:0	TO31-TO0	Признак завершения передачи сообщения из соответствующего элемента Tx Buff.	R	0x0

37.11.25 TXBCF - Регистр подтверждения отмены передачи буфера Tx Buffer

Отражает состояние завершения отмены передачи для буфера Tx Buff. Бит TXBCF устанавливается, когда сбрасывается соответствующий бит TXBRP в результате запроса отмены передачи TXBCR или отмены по таймауту (см. TS64). Бит TXBCF сбрасывается, когда происходит запись «1» в соответствующий разряд TXBAR. Формат регистра TXBCF приведён в следующей таблице:

Таблица 37.72. Формат регистра TXBCF

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:0	CF31-CF0	Признак подтверждения отмены передачи сообщения из соответствующего элемента Tx Buff.	R	0x0

37.11.26 TXBTIE - Регистр разрешения прерывания по окончании передачи буфера Tx Buffer

Запись «1» в любые разряды разрешает прерывание IR.TC при передаче из соответствующих буферов Tx Buff. Формат регистра TXBTIE приведён в следующей таблице:

Таблица 37.73. Формат регистра TXBTIE

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:0	TIE31-TIE0	Разрешения прерывание при передаче сообщения от соответствующего буфера Tx Buff.	RW	0x0

37.11.27 TXBCIE - Регистр разрешения прерывания при подтверждении отмены передачи буфера Tx Buffer

Запись «1» в любые разряды разрешает прерывание IR.TCF при успешной отмене передачи для соответствующих буферов Tx Buff. Формат регистра TXBCIE приведён в следующей таблице:

Таблица 37.74. Формат регистра TXBCIE

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:0	CFIE31-CFIE0	Разрешения прерывание при успешной отмене передачи сообщения для соответствующего буфера Tx Buff.	RW	0x0

37.11.28 TXEFC - Регистр конфигурации буфера Tx Event FIFO

Формат регистра TXEFC приведён в следующей таблице:

Таблица 37.75. Формат регистра TXEFC

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:30	-	Резерв	R	0x0
29:24	EFWM	Значение Watermark определяет количество данных в буфере Tx Event FIFO при котором сформируется прерывание IR.TEFW: 0 - Watermark прерывание отключено; 1-64 - значение watermark Tx Event FIFO; >64 - Watermark прерывание отключено.	RP	0x0
23:0	-	Резерв	R	0x0

37.11.29 TXEFS - Регистр статуса буфера Tx Event FIFO

Отражает статус буфера Tx Event FIFO. Формат регистра TXEFS приведён в следующей таблице:

Таблица 37.76. Формат регистра TXEFS

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:26	-	Резерв	R	0x0
25	TEFL	Признак потери элемента Tx Event FIFO: 0 - Потери элемента нет; 1 - Новый элемент не записан в Tx Event FIFO, так как Tx Event FIFO заполнено. Признак совпадает с сигналом IR.TEFL.	R	0x0
24	EFF	Признак заполненного буфера Tx Event FIFO: 0 - буфер не полон; 1 - буфер полон.	R	0x0
23:6	-	Резерв	R	0x0
5:0	EFFL	Количество сообщений в буфере Tx Event FIFO (0 - 31).	R	0x0

37.11.30 TTRMC - Регистр конфигурации опорного сообщения

Формат регистра TTRMC приведён в следующей таблице:

Таблица 37.77. Формат регистра TTRMC

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31	-	Резерв	R	0x0
30	XTD	Тип идентификатора опорного сообщения: 0 - 11-бит стандартный идентификатор; 1 - 29-бит расширенный идентификатор.	RP	0x0
29	-	Резерв	R	0x0
28:0	RID[28:0]	Идентификатор опорного сообщения и используемый для фильтрации. Бит XTD определяет тип идентификатора опорного сообщения стандартный или расширенный. Стандартный идентификатор содержится в ID[10:0].	R	0x0

37.11.31 TTOCF - Регистр конфигурации режима планирования событий

Формат регистра TTOCF приведён в следующей таблице:

Таблица 37.78. Формат регистра TTOCF

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:27	-	Резерв	R	0x0
26	EVTP	Полярность Event Trigger: 0 - передний фронт; 1 - задний фронт. * В данной реализации не используется *	RP	0x0

25	ECC	Автоматическая калибровка шага времени для Time Slave или резервного Time Master: 0 - отключена; 1 — включена. * В данной реализации не используется *	RP	0x0
24	-	Резерв	R	0x0
23:16	AWL[7:0]	Порог срабатывания Watchdog таймера. Watchdog таймер инкременируется каждые 256 единиц локального времени. Таймер watchdog обслуживается чтением регистра TTOST. Если таймер watchdog не обслужен до порога срабатывания, устанавливается флаг TTOST.AWE, передача прекращается, и контроллер переходит в режим монитора. Таймер Watchdog можно отключить записью TTOCF.AWL = 0x00. * В данной реализации не используется *	RP	0x1
15	EECS	Разрешение корректировки шага времени для текущего Time Master: 0 - запрещено; 1 — разрешено. * В данной реализации не используется *	RP	0x0
14:8	IRTO[6:0]	Начальное смещение опорного сообщения, складывается со значением времени метки Ref_Trigger(_Gap). (см. п. 37.12.12). 0x00-7F положительное смещение, в диапазоне от 0 до 127.	RP	0x0
7:5	LDSDL[2:0]	Граница разброса синхронизации SDL задаётся через логарифм LDSDL по основанию 2. SDL = 2(LDSDL + 5). (см. п. 37.12.12.3). 0x0-7 -> SDL ≤ 32…4096. * В данной реализации не используется *	RP	0x0
4	TM	Выбор режима time master/time slave: 0 - Time Slave; 1 - потенциальный Time Master.	RP	0x0
3	GEN	Разрешение пауз (Gap): 0 - запрещено; 1 - разрешено.	RP	0x0
2	-	Резерв	R	0x0
1:0	OM[1:0]	Включение режима планирования событий: 00 - режим планирования событий отключён, по умолчанию; 01 - режим планирования событий Level1; 10 - режим планирования событий Level2; 11 - не используется.	RP	0x0

37.11.32 TTMLM - Регистр конфигурации системной матрицы

Формат регистра TTMLM приведён в следующей таблице:

Таблица 37.79. Формат регистра TTMLM

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:12	-	Резерв	R	0x0
11:8	TXEW[3:0]	Длительность периода разрешения передачи сообщения (Tx Enable): 0x0-F , единиц NTU.	RP	0x0

7:6	CCS[1:0]	Условие формирования импульса на выходе SYNC_OUT: 00 – не формируется; 01 – импульс при каждом опорном сообщении; 10 – импульс при опорном сообщении, которое завершает цикл системной матрицы (TTCTC.CC=TTNLN.CCM); 11 – не формируется. * В данной реализации не используется *	RP	0x0
5:0	CCM[5:0]	Количество циклов в системной матрице: 0x00 1 цикл; 0x01 2 цикла; 0x03 4 цикла; 0x07 8 циклов; 0x0F 16 циклов; 0x1F 32 циклов; 0x3F 64 циклов; остальные значения зарезервированы.	RP	0x0

37.11.33 TURCF - Регистр конфигурации делителя TUR

Более подробная информация приведена в следующей таблице:

Таблица 37.80. Формат регистра TURCF

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31	ELT	Управление локальным временем: 0 - Локальное время остановлено; 1 - Локальное время запущено.	RP	0x0
30	-	Резерв	R	0x0
29:16	DC[13:0]	Делитель для вычисления значения TUR: 0x0000 - не допустимое значение; 0x0001-3FFF - значение делителя.	RP	0x1000
15:0	NCL[15:0]	Множитель для вычисления значения TUR. Запись в поле NCL возможна при TURCF.ELT = “0” или TTOCF.EECS = “1”. Если новое значение записывается не в режиме конфигурации, оно будет применено после сброса TTOST.WECS. Поле NCL защищено от записи при TTOST.WECS = “1”. 0x0000-FFFF значение младшей части множителя.	RP	0x0

37.11.34 TTOCN - Регистр управления режимом планирования событий

Формат регистра TTOCN приведён в следующей таблице:

Таблица 37.81. Формат регистра TTOCN

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:13	-	Резерв	R	0x0
12	NIG	Установка паузы (GAP). Разряд может устанавливаться только в текущем time master работающем в режиме внешней синхронизации (TTOCF.GEN = “1”). После записи “1” в поле NIG следующее опорное сообщение будет содержит Next_is_Gap = “1”, при этом значение бита NIG станет = 0.	RW	0x0
11	-	Резерв	R	0x0

10	FGP	Остановка паузы (GAP). После записи “1” в поле FGP инициируется передача опорного сообщения (завершение паузы). После передачи опорного сообщения бит FGP сбросится.	RW	0x0
9	GCS	Внешнее управление паузой (GAP): 0 - Gap независит от EXT_EVENT; 1 - Gap управляется входом EXT_EVENT.	RW	0x0
8:2	-	Резерв	R	0x0
1	ECS	Коррекция шага локального времени. Запись “1” в ECS установит TTOST.WECS если узел является текущим Time Master. ECS сбросится на следующем такте. Значение TURCF.NC запишется в TURNA.NAV в начале следующего цикла. * В данной реализации не используется *	RW	0x0
0	-	Резерв	R	0x0

37.11.35 TTIR - Регистр прерываний в режиме планирования событий

Каждое прерывание может быть разрешено отдельно соответствующим битом в регистре TTIE. Флаги остаются установленными пока не будут очищены записью ”1” в соответствующий разряд.

Флаги CER, AW и WT отражают состояние конфигурационной ошибки. Эти ошибки требуют реконфигурации контроллера перед началом повторного обмена. * В данной реализации CER, AW не используются *

Флаг GTE отражает состояние ошибок. Если они вызваны возмущениями на CAN шине, они будут обработаны CAN протоколом и не потребуют участия со стороны ПО. * В данной реализации не используются *

Флаги SOG, SMC и SBC обеспечивают синхронизацию с планировщиком.

Флаги коммутируются по логическому «или» на INT1. Формат регистра TTIR приведён в следующей таблице:

Таблица 37.82. Формат регистра TTIR

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:19	-	Резерв	R	0x0
18	CER	Ошибка конфигурации списка элементов событий.	RW	0x0
17	AW	Ошибка watchdog таймера. * В данной реализации не используется *	RW	0x0
16	WT	Ошибка Watch trigger. Прерывание устанавливается если выполняется Watch trigger, что означает отсутствие опорного сообщения в данном цикле.	RW	0x0
15:10	-	Резерв	R	0x0
9	GTE	Прерывание происходит если значение девиации синхронизации (SD) превышает значение определяемое полем TTOCF.LDSDL. * В данной реализации CER, AW не используются *	RW	0x0
8:4	-	Резерв	R	0x0

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
3	SOG	Начало паузы (GAP). Прерывание происходит когда обнаруживается опорное сообщений с Next_is_Gap = '1'.	RW	0x0
2	-	Резерв	R	0x0
1	SMC	Начало нового цикла матрицы. Прерывание происходит когда счётчик циклов достигает значения TTMLM.CCM.	RW	0x0
0	SBC	Начало нового цикла. Прерывание происходит когда счётчик циклов увеличивается на 1.	RW	0x0

37.11.36 TTIE - Регистр разрешения прерываний в режиме планирования событий

Разрешение для прерываний в регистре TTIR. Запись «1» разрешает соответствующее прерывание. Формат регистра TTIE приведён в следующей таблице:

Таблица 37.83. Формат регистра TTIE

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:19	-	Резерв	R	0x0
18	CERE	Разрешение прерывания в случае конфигурационной ошибки. * В данной реализации не используется *	RW	0x0
17	AWE	Разрешение прерывания в случае ошибки watchdog таймера. * В данной реализации не используется *	RW	0x0
16	WTE	Разрешение прерывания в случае события Watch Trigger.	RW	0x0
15:10	-	Резерв	R	0x0
9	GTEE	Разрешение прерывания в при превышении девиации синхронизации значения определяемого полем TTOCF.LDSDL. * В данной реализации не используется *	RW	0x0
8:4	-	Резерв	R	0x0
3	SOGE	Разрешение прерывания в случае начала паузы (GAP).	RW	0x0
2	-	Резерв	R	0x0
1	SMCE	Разрешение прерывания в начале нового цикла матрицы.	RW	0x0
0	SBCE	Разрешение прерывания в начале нового цикла.	RW	0x0

37.11.37 TTOST - Регистр статуса режима планирования событий

Формат регистра TTOST приведён в следующей таблице:

Таблица 37.84. Формат регистра TTOST

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31	-	Резерв	R	0x0
30	WECS	Ожидание коррекции шага времени (значение TUR). Бит сбросится с наступлением следующего цикла. * В данной реализации не используется *	R	0x0
29	AWE	Статус ошибки watchdog таймера. * В данной реализации не используется *	R	0x0
28	WFE	Отражает состояние ожидания паузы (GAP).	R	0x0
27	GSI	Отражает состояние паузы (GAP).	R	0x0
26:24	TMP[2:0]	Приоритет текущего time master.	R	0x0

23	GFI	Отражает состояние завершения паузы (GAP). Устанавливается при TTOCN.FGP=”1” или при активном входе EXT_EVENT если TTOCN.GCS= “1”. Бит сбросится с наступлением следующего цикла.	R	0x0
22:16	-	Резерв	R	0x0
15:8	RTO[7:0]	Смещение опорного сообщения (-127.+127)	R	0x0
7	QCS	Статус девиации синхронизации (для режима Level2, в остальных случаях = “1”): 0 - локальное время не синхронизировано с Time Master; 1 - отклонение синхронизации в пределах SDL. * В данной реализации не используется * При чтении возвращает 1.	R	0x1
6	-	Резерв	R	0x0
5:4	SYS[1:0]	Статус состояния синхронизации: 00 - Не синхронизирован; 01 - Синхронизация планировщика; 10 - Планировщик в паузе (In_Gap); 11 - Планировщик в нормальном режиме (In_Schedule).	R	0x0
3:2	MS[1:0]	Статус Time master: 00 - не инициализировано; 01 - Контроллер в состоянии Time Slave; 10 - Контроллер в состоянии резервного Time Master; 11 - Контроллер в состоянии текущего Time Master.	R	0x0
1:0	EL[1:0]	Статус режима планирования событий: 00 - нет ошибки; 01 – не используется; 10 – не используется; 11 - конфигурационная ошибка.	R	0x0

37.11.38 TURNA - Регистр текущего значения делителя TUR

В режиме Level1 (NAV = NC) компенсации смещения нет. В режиме Level2, вычисляется смещение тактов между узлом, принявшим опорное сообщение и узлом, отправившим его. Смещение компенсируется когда отклонение не превышает значения 1<<(TTOCF.LDSDL + 5).

Формат регистра TURNA приведён в следующей таблице:

Таблица 37.85. Формат регистра TURNA

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:18	-	Резерв	R	0x0
17:0	NAV	0x0F000-20FFF текущее значение числителя.	R	0x0

37.11.39 TTLGT - Регистр локального и глобального времени

Назначение регистра описано в п. 37.12.12.7.Формат регистра TTLGT приведён в следующей таблице:

Таблица 37.86. Формат регистра TTLGT

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:16	GT	Глобальное время.	R	0x0

15:0	LT	Локальное время.	R	0x0

37.11.40 TTCTC - Регистр времени цикла и номера цикла

Формат регистра TTCTC приведён в следующей таблице:

Таблица 37.87. Формат регистра TTCTC

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:22	-	Резерв	R	0x0
21:16	CC	Количество циклов в системной матрице.	R	0x0
15:0	CT	Время цикла.	R	0x0

37.11.41 STATX - Регистр числа ошибок при передаче

Формат регистра STATTX приведён в следующей таблице:

Таблица 37.88. Формат регистра STATTX

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:0	txerr_cnt	Число ошибок при передаче сообщений (кроме ошибок арбитража). Увеличивается на единицу в начале кадра ошибки, выдаваемого после обнаружения ошибки во время передачи сообщения. После достижения максимального значения 0xFFFFFFFF подсчёт прекращается, регистр сохраняет своё значение. Для продолжения подсчёта необходимо записать в регистр требуемое начальное значение, счёт продолжится с этого значения.	RW	0x0

37.11.42 STATRX - Регистр числа ошибок при приёме

Формат регистра STATRX приведён в следующей таблице:

Таблица 37.89. Формат регистра STATRX

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:0	rxerr_cnt	Число ошибок при приёме сообщений (кроме ошибок арбитража). Увеличивается на единицу в начале кадра ошибки, выдаваемого после обнаружения ошибки при приёме сообщения. После достижения максимального значения 0xFFFFFFFF подсчёт прекращается, регистр сохраняет своё значение. Для продолжения подсчёта необходимо записать в регистр требуемое начальное значение, счёт продолжится с этого значения.	RW	0x0

37.11.43 STATWRN - Регистр числа превышений уровня предупреждения

Формат регистра STATWRN приведён в следующей таблице:

Таблица 37.90. Формат регистра STATWRN

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:0	wrn_cnt	Число превышений счётчиками TEC и REC уровня предупреждения (>=96 ошибок)шибок при передаче сообщений(кроме ошибок арбитража). После достижения максимального значения 0xFFFFFFFF подсчёт прекращается, регистр сохраняет своё значение. Для продолжения подсчёта необходимо записать в регистр требуемое начальное значение, счёт продолжится с этого значения.	RW	0x0

37.11.44 STATEP - Регистр числа переходов в состояние Error Passive

Формат регистра STATEP приведён в следующей таблице:

Таблица 37.91. Формат регистра STATEP

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:0	ep_cnt	Число переходов в состояние Error Passive. После достижения максимального значения 0xFFFFFFFF подсчёт прекращается, регистр сохраняет своё значение. Для продолжения подсчёта необходимо записать в регистр требуемое начальное значение, счёт продолжится с этого значения.	RW	0x0

37.11.45 STATBO - Регистр числа переходов в состояние Bus_Off

Формат регистра STATBO приведён в следующей таблице:

Таблица 37.92. Формат регистра STATBO

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:0	bo_cnt	Число переходов в сотояние Bus_Off. После достижения максимального значения 0xFFFFFFFF подсчёт прекращается, регистр сохраняет своё значение. Для продолжения подсчёта необходимо записать в регистр требуемое начальное значение, счёт продолжится с этого значения.	RW	0x0

37.11.46 STATLOST - Регистр числа потерянных сообщений при приёме

Формат регистра STATLOST приведён в следующей таблице:

Таблица 37.93. Формат регистра STATBO

Номер разряда	Условное обозначение	Назначение	Доступ	Исходное состояние
31:0	lost_cnt	Число потерянных сообщений из-за полной заполненности приёмного буфера. После достижения максимального значения 0xFFFFFFFF подсчёт прекращается, регистр сохраняет своё значение. Для продолжения подсчёта необходимо записать в регистр требуемое начальное значение, счёт продолжится с этого значения.	RW	0x0

37.12 Функциональное описание CAN

37.12.1 Программная инициализация

Переход в режим программной инициализации происходит при установке бита CCCR.INIT= “1” или при аппаратном сбросе. При CCCR.INIT=”1” обмен по CAN шине не происходит и выход находится в рецессивном состоянии. Конфигурационные регистры остаются неизменными при переходе в режим инициализации. После сброса CCCR.INIT=”0” ядро контроллера ожидает 11 рецессивных бит на CAN шине для перехода в режим ожидания и возобновления обмена по шине.

Доступ регистрам с типом доступа «RP» осуществляется при установленных CCCR.INIT = “1” и CCCR.CCE = “1”. Поле CCCR.CCE можно изменять при CCCR.INIT = “1”, CCCR.CCE автоматически сбрасывается при CCCR.INIT = “0”

Следующие регистры сбрасываются при CCCR.CCE = “0”:

- RXF0S;
- TXBRP;
- TXBTO;
- TXBCF;
- TXEFS.

Следующие регистры доступны для записи только при CCCR.CCE = “0”:

- TXBAR;
- TXBCR.

CCCR.DAR и CCCR.MON доступны для записи только при CCCR.INIT = “1” и CCCR.CCE=”1”.

Контроллер выходит их режима инициализации после сброса CCCR.INIT = “0”.

37.12.2 Отключение автоматического повтора передачи.

Контроллер поддерживает автоматический повтор передачи сообщений проигравших арбитраж или отложенных из-за ошибок при передаче. По умолчанию автоматический повтор передачи включён. Автоматический повтор отключается через CCCR.DAR.

37.12.3 Режим монитора

Контроллер переводится в режим монитора установкой бита CCCR.MON= “1”. В этом режиме на шину CAN выдаются только рецессивные биты. Если требуется отправить доминантный бит (флаг acknowledge, флаг ошибки), бит перенаправляется внутри таким образом, что контроллер протокола видит этот доминантный бит, в то время как CAN шина может оставаться в рецессивном состоянии. В режиме монитора регистр TXBRP удерживается в состоянии сброса. Режим монитора может использоваться для анализа трафика на CAN шине, без воздействия на неё доминантными битами. Кадр будет принят только после успешного его завершения. Также будут приниматься кадры от передатчиков, находящихся в состоянии Error Passive, даже в том случае, если на них не было признака подтверждения ACK. Значение поля MON_AE элемента Rx FIFO, указывает на наличие/отсутствие ошибки подтверждения.

37.12.4 Режим Loop Back

Перевод в данный режим осуществляется записью CCCR.LBM= “1”. В режиме Loop Back, исходящие сообщения зацикливаются и сохраняются (в случае соответствия фильтрам) в памяти. Это обеспечивает возможность самотестирования аппаратной части. Будучи независимым от внешнего воздействия, контроллер игнорирует ошибки acknowledge в режиме Loop Back. В этом режиме вход RX отключается от CAN шины.

37.12.5 Режим внутренний Loop Back

Перевод в данный режим осуществляется записью CCCR.LBM= “1” и CCCR.MON= “1”. В режиме внутреннего Loop Back контроллер полностью отключается от шины, исходящие сообщения зацикливаются и сохраняются (в случае соответствия фильтрам) в памяти.

Будучи независимым от внешнего воздействия, контроллер игнорирует ошибки acknowledge в режиме Loop Back. В этом режиме вход RX и выход TX отключается от CAN шины.

37.12.6 Режим внешний Loop Back

Перевод в данный режим осуществляется записью CCCR.LBM= “1” и CCCR.MON= “0”. В

этом режиме, на выходе TX передаваемые данные будут появляться, но состояние входа RX игнорируется. Внутри контроллера TX подключается к RX.

На следующем рисунке показаны схема подключения входных/выходных сигналов CAN в зависимости от режима LoopBack и монитор.

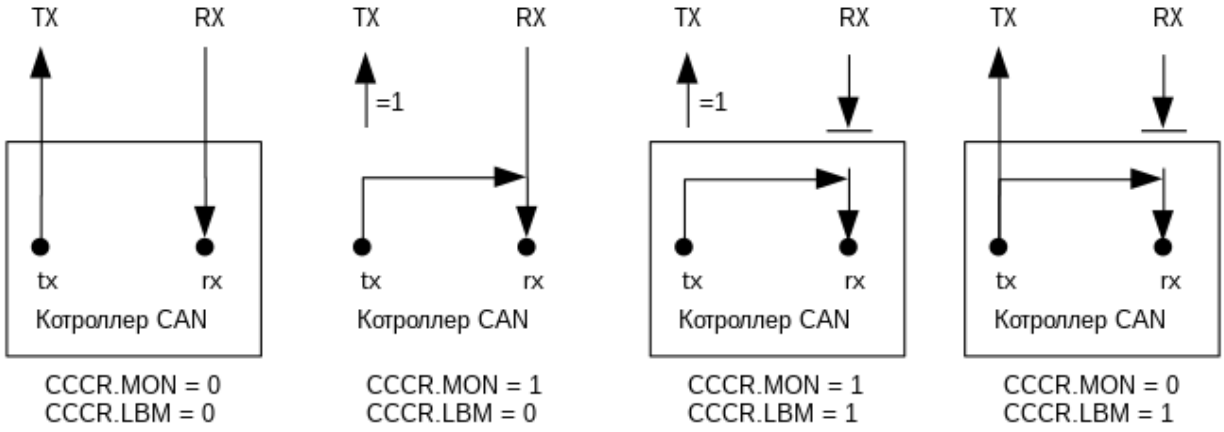


Рисунок 37.41. Комбинаций режимов LoopBack и монитор

37.12.7 Временные метки (Timestamp)

Для генерации временных меток в контроллере CAN используется 16-ти разрядный циклический Timestamp счётчик. Делитель TSCC.TCP определяет частоту тактирования счётчика в диапазоне 1-16 битовых интервалов шины. Значение счётчика доступно для чтения в поле TSCV.TSC. Запись в регистр TSCV сбрасывает значение счётчика в 0. Когда счётчик переполняется выставляется флаг прерывания IR.TSW.

В начале принимаемого/передаваемого кадра, текущее значение счётчика помещается в поля RXTS[15:0] или TXTS[15:0]. Установкой бита TSCC.TSS можно переключится на внешнее тактирование.

37.12.8 Watchdog таймер

Контроллер содержит watchdog таймер для контроля функционирования. Таймер необходимо постоянно опрашивать, в противном случае вся активность на CAN шине прекращается. Значение watchdog таймера TTOCF.AWL описывает длительность (в NTU) промежутка времени в течении которого он должен быть обслужен. Максимальное значение этого параметра – 256 NTU. Сторожевой таймер обслуживается чтением регистра TTOST. TTOST.AWE отражает был ли watchdog таймер обслужен вовремя. Если произошел сбой при обслуживании watchdog таймера, устанавливается флаг прерывания TTIR.AW. Watchdog таймер может быть отключён записью 0x00 в поле TTOCF.AWL. * В данной реализации не используется *

37.12.9 Прием сообщения

Модуль управления приёмом отвечает за фильтрацию и передачу принятых кадров в буфер Rx FIFO. Формат буфера Rx FIFO в п 37.10.3. Формат фильтров описан в Разделах 37.10.1 и 37.10.2.

37.12.9.1 Входная фильтрация

В контроллере организовано два набора приёмных фильтров, один для стандартных идентификаторов (11-бит) и один для расширенных идентификаторов (29-бит). Прошедшие фильтрацию сообщения сохраняются в буфер Rx FIFO. Каждый набор фильтров исполняется с #0 элемента и до первого соответствующего элемента (или конца списка). Следующий фильтра уже не выполняется для текущего сообщения.

Конфигурация фильтров:

- Каждый фильтр может быть сконфигурирован как:
  - диапазон идентификаторов (от — до),
  - фильтр на 1 или 2 выделенных идентификатора,
  - битовая маска;
- Каждый фильтр может принимать или отклонять кадры;
- Каждый фильтр включается/выключается индивидуально;
- Фильтры выполняются последовательно до первого совпадающего элемента.

Используемые конфигурационные регистры:

- SIDFC;
- XIDFC;
- XIDAM.

В зависимости от конфигурации фильтра (SFEC/EFEC), соответствие запускает одно из следующих действий:

- сохранение принятого кадра в Rx FIFO буфере;
- отклонение принятого кадра.

37.12.9.2 Фильтр диапазона идентификаторов

Идентификаторы в принимаемых кадрах проверяются на попадание в допустимый диапазон, заданный в регистрах SF1ID/SF2ID (стандартный ID), EF1ID/EF2ID (расширенный ID).

Возможно два способа использования фильтра для расширенных кадров:

EFT = “00”: Идентификатор в принимаемых кадров складывается по логическому «и» с маской (XIDAM) до применения диапазонного фильтра

EFT = “11”: дополнительная маска (XIDAM) не используется.

37.12.9.3 Фильтр отдельного идентификатора

Элемент фильтра может быть сконфигурирован для фильтрации 1 или 2 отдельных идентификаторов. Для фильтрации одного идентификатора, в конфигурации стандартного элемента фильтра должно быть задано SF1ID = SF2ID, в расширенном EF1ID = EF2ID.

37.12.9.4 Фильтр битовая маска

Битовая маска используется для фильтрации групп идентификаторов маскированием единичных бит идентификаторов принимаемых кадров. При фильтрации битовой маской SF1ID/EF1ID используется как фильтр идентификаторов, а SF2ID/EF2ID используется как маска.

Бит маски фильтра со значением «0» замаскирует соответствующую битовую позицию сконфигурированного фильтра, т.е. значения этих битов в принятом идентификаторе не будут иметь значения при фильтрации.

Только те биты идентификатора в принятом кадре будут участвовать в фильтрации, позиция которых совпадает с позицией бит маски со значением «1».

Когда все биты маски равны «1», то идентификатор должен полностью совпадать со значением фильтра, если все биты маски равны «0», то любой идентификатор будет проходить фильтрацию.

37.12.9.5 Фильтрация сообщений

Идентификатор кадра (ID), бит удалённого запроса (RTR), и бит расширения идентификатора (IDE), принятых кадров, сравниваются с набором сконфигурированных элементов фильтра, под управлением регистров GFC, SIDFC или XIDFC. Расширенный ID принятого кадра складывается по логическому «и» с маской из регистра XIDAM перед использованием фильтра.

37.12.9.6 Буфер приёма Rx FIFO

Буфер Rx FIFO может содержать до 64 элементов. Формат элементов буфера Rx FIFO описан в пункте 37.10.3. Конфигурация буфера осуществляется через регистр RXF0C.

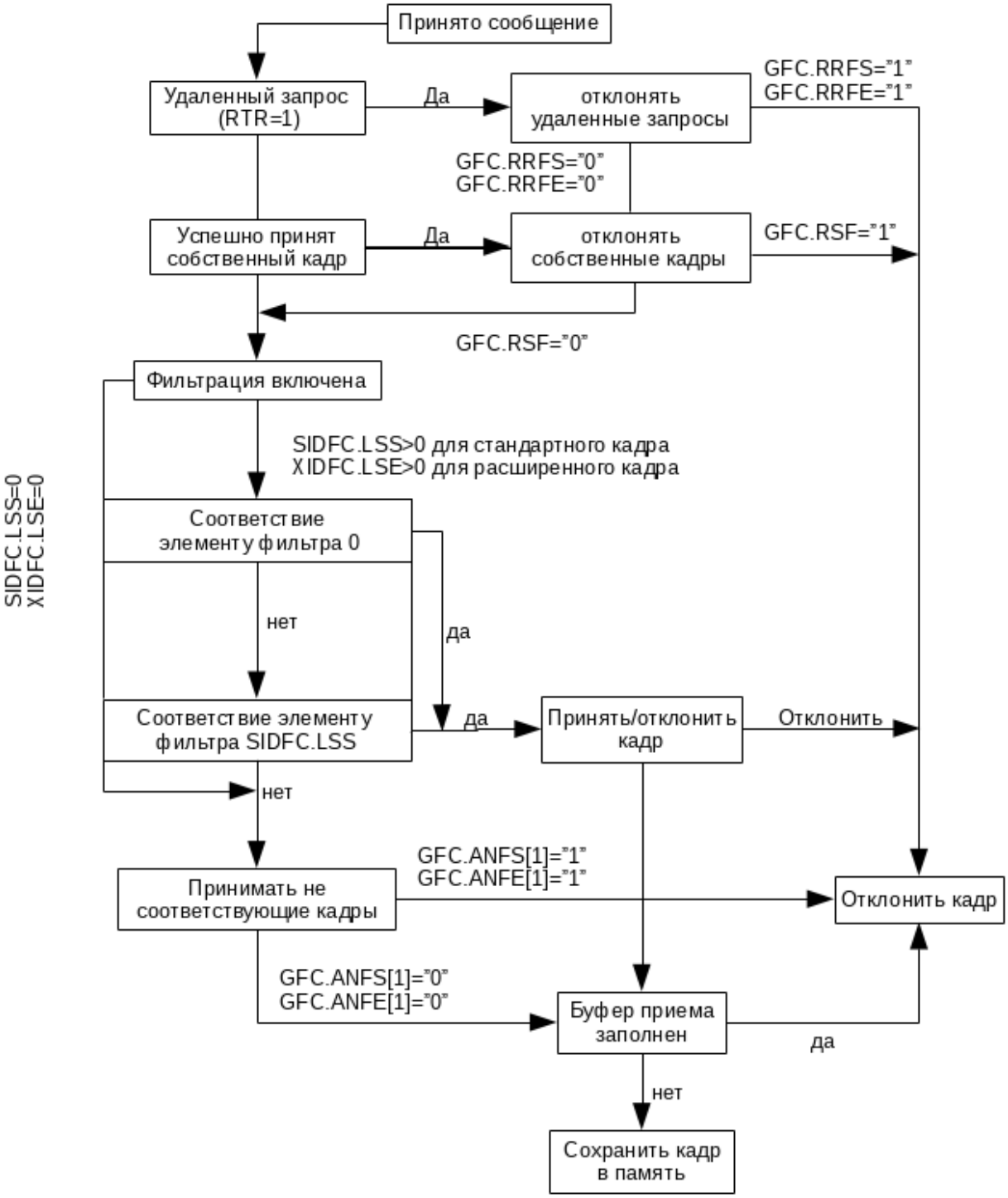


Рисунок 37.42. Схема фильтрации входящих сообщений

Прошедшие входную фильтрацию кадры, помещаются FIFO буфер, при соответствующих настройках поля SFEC/EFEC фильтров. Когда буфер FIFO полностью заполнится, выставляется флаг IR.RF0F, больше в этот буфер сообщения не помещаются пока хотя бы одно не будет прочитано. Если сообщение принято, при заполненном буфере – сообщение отклоняется и устанавливается флаг прерывания IR.RF0L.

Во избежание переполнения буфера Rx FIFO, следует использовать метку заполнения. Когда количество элементов Rx FIFO достигает значения заданного в RXF0C.F0WM выставляется флаг прерывания IR.RF0W, и остаётся установленным пока количество элементов в буфере не будет меньше указанного значения.

Чтение из буфера Rx FIFO следует производить по начальному адресу буфера Rx FIFO необходимое количество раз. При каждом обращении читается 32-битное слово. В регистре RXF0S отображается состояние буфера.

37.12.10 Передача сообщения

Передача сообщения осуществляется при установке в «1» разряда регистра TXBAR соответствующего номеру буфера передачи. Формат буферов передачи описан в Разделе 37.10.5.

Если TXBRP не равен 0 и не происходит отправка сообщения, происходит сканирование буферов передачи, выбирается буфер с наивысшим приоритетом (см. TXBC.PRI_MODE).

37.12.10.1 Буферы Tx Buff

В каждом буфере передачи может храниться одно сообщение. Если несколько буферов содержат сообщения с одинаковыми приоритетами, то для передачи выбирается буфер с наименьшим порядковым номером. Формат элементов буфера Tx Buff описан в пункте 37.10.5.

Запрос передачи, при появлении новых данных, осуществляется через регистр TXBAR.ARn.

37.12.10.2 Отмена передачи

Для отмены запроса на передачу необходимо записать «1» в соответствующий разряд (равный порядковому номеру буфера передачи) регистра TXBCR.

При успешной отмене передачи, соответствующий разряд регистра TXBCF устанавливается в «1». В случае если запрос на отмену передачи произошёл во время передачи, соответствующий разряд TXBRP остаётся установленным в «1» до окончания передачи. Если передача была успешна, соответствующие разряды регистров TXBTO и TXBCF устанавливаются в «1». Если передача не удалась, то повторной передачи не происходит и соответствующий разряд регистра TXBCF устанавливается в «1».

37.12.10.3 Буфер Tx Event FIFO

После успешной передачи сообщения, контроллер сохраняет идентификатор переданного сообщения и временную метку (timestamp) в буфере Tx Event FIFO, если установлен соответствующий бит (EFC) в элементе буфера передачи. Формат элементов буфера Tx Event FIFO описан в пункте 37.10.4.

Чтобы «связать» событие отправки сообщения с элементами буфера Tx Event FIFO, поле MM (Маркер сообщения) в переданном сообщении, копируется в элемент буфера Tx Event FIFO. Буфер Tx Event FIFO может содержать до 32 элементов.

Когда буфер Tx Event FIFO полностью заполнен, устанавливается флаг IR.TEFF, последующие сообщения не помещаются в буфер Tx Event FIFO пока из него не будет прочитан хотя бы один элемент. В случае попытки записи элемента в заполненный буфер Tx Event FIFO, запись будет отменена и будет установлен флаг в регистре прерываний IR.TEFL.

Во избежание переполнения буфера Tx Event FIFO, следует использовать метку заполнения. Когда количество элементов FIFO достигает значения заданного в TXEFC.EFWM выставляется флаг прерывания IR.TEFW, и остаётся установленным пока количество элементов в буфере не будет меньше указанного значения.

Чтение из буфера Tx Event FIFO следует производить по начальному адресу буфера Tx Event FIFO необходимое количество раз. При каждом обращении читается 32-битное слово. В регистре TXEFS отображается состояние буфера.

37.12.11 Режим Bus_Off

При входе в режим Bus_Off (счётчик ошибок передачи ECR.TEC достиг значения большего 255, см. спецификацию CAN) выдача запрошенных сообщений приостанавливается, взводится признак PSR.BO, устанавливается прерывание IR.BO (если разрешено), устанавливается CCCR.INIT, сбрасывается счётчик ECR.REC.

При входе в режим Bus_Off счётчик ECR.TEC устанавливается в значение 128, после чего начинает декрементироваться на единицу при каждом обнаружении 11 последовательных рецессивных битов в шине. После того, как ECR.TEC достигает значения 0, формируется необходимое, но недостаточное условие выхода из режима Bus_Off.

Для выхода из режима Bus_Off должны быть выполнены два условия:

Счётчик ECR.TEC достиг значения 0 (в шине было обнаружено 128 последовательностей по 11 непрерывно идущих рецессивных битов)

Программно сброшен бит CCCR.INIT.

Продолжить передачу заданных сообщений (см. TXBRP, TXBTO и др) можно просто программно сбросив бит CCCR.INIT. Если CCCR.INIT был сброшен раньше достижения счётчиком ECR.TEC нуля, передача начнётся только после достижения нуля счётчиком ECR.TEC.

Если при выходе из Bus_Off не нужно продолжать передачу заданных ранее сообщений, необходимо установить CCCR.CCE, что приведёт с сбросу регистров TXBRP, TXBTO и др.,

после чего сбросить бит CCCR.INIT. Далее запуск необходимых сообщений производится обычным способом (запись в TXBAR).

37.12.12 Режим планирования событий

Режим планирования событий позволяет синхронизировать работу нескольких узлов CAN. Конфигурируемые элементы событий, управляют коммуникацией на шине позволяя детерминировать время отправки сообщений и синхронизировать отсчёт времени контроллера с помощью опорных сообщений. Режим включается, когда поле TTOCF.OM отлично от 0.

Поле TTOCF.TM определяет режим работы контроллера (как потенциальный time master или как time slave). Для режима time master три младших бита идентификатора опорного сообщения (TTRMC.RID) отражают приоритет мастера, 0 наивысший, 7 низший. На шине не должно быть двух мастеров с одинаковым приоритетом. Поле TTRMC.RID используется для распознавания опорных сообщений.

Поле TTOCF.IRTO (Начальное смещение опорного сообщения) это 7-битное значение которое описывает (в NTU) время, которое резервный time master ожидает перед перед тем, как начать передачу опорного сообщения, относительно метки времени опорного сообщения. Рекомендованное значение TTOCF.IRTO должно быть пропорционально приоритету мастера. Последовательность, в которой резервный time master сменяет текущий и отправляет опорное сообщение, должно определяться их приоритетом даже в случае значительного смещения тактов.

Поле TTOCF.OM определяет вид режима планирования событий – Level1 или Level2:

TTOCF.OM = “00” Узел функционирует в соответствии с ISO 11898-1, режим планирования событий отключён.

TTOCF.OM = “01” (Level1) Узел функционирует в соответствии с ISO 11898-1, с поддержкой режима планирования событий, но без возможности синхронизации передачи опорных сообщений по внешним событиям, бит Next_is_Gap в опорном сообщении игнорируется.

TTOCF.OM = “10” (Level2) Узел функционирует в соответствии с ISO 11898-1, с поддержкой режима планирования событий, включая синхронизацию передачи опорных сообщений по внешнему событию.

Поле TTOCF.EECS разрешает внешнюю синхронизацию, позволяя программно обновить конфигурацию TUR на текущем time master (только в Level2).

Поле TTMLM.CCM указывает номер последнего цикла в системной матрице. Отсчёт циклов начинается с 0. В системной матрице, содержащей 8 циклов TTMLM.CCM, будет 7.

Поле TTMLM.TXEW указывает длину окна Tx enable в NTU. Окно Tx enable это период времени в начале временного окна в течение которого может начаться передача. Если начало передаваемого сообщения выходит за границу Tx enable (например, из-за наложения сообщения из предыдущего окна), передача не произойдёт. TTMLM.TXEW должен быть выбран исходя из качества синхронизации сети и соотношения размера окна и размера сообщения.

37.12.12.1 Опорное сообщение

Принимается всеми узлами, кроме узла отправившего сообщение (текущий Time master, см. п. 37.12.12.2). Длина поля данных опорного сообщения в Level1 не менее 1 байта. Длина опорного сообщения в Level2 не более 4 байт, в противном случае сообщение не распознаётся как опорное. Младшие три бита идентификатора отражают приоритет time master, до 8 потенциальных time master могут быть на одной шине. Опорное сообщение конфигурируется через регистр TTRMC.

Если передача опорного сообщения прервана оно немедленно передаётся снова и в случае повторной передачи поле Master_Ref_Mark обновляется. Опорное сообщение отправляется периодически если не установлен бит Next_is_Gap.

Если текущий time master перестал передавать опорное сообщение его функцию начинает выполнять один из потенциальных time master.

- Режим Level2

Активен при TTOCF.OM = “10” и TTOCF.GEN = “1”. Информация, относящаяся к опорному сообщению, содержится в первых 4 байтах поля данных (32-бита). Формат поля данных опорного сообщения для режима Level2 приведён в следующей таблице:

Таблица 37.94. Формат поля данных опорного сообщения для режима Level2

Номер разряда	Условное обозначение	Назначение
31:16	Master_Ref_Mark	Значение локального времени текущего time master в момент отправки опорного сообщения.
15:8	-	Резерв
7	NIG	Next is GAP.
6	-	Резерв
5:0	CC	Номер цикла текущего time master.

- Режим Level1

Один из режимов синхронизации узлов CAN сети в режиме планирования событий. Активен при TTOCF.OM = “01” и TTOCF.GEN = “0”. Внешняя синхронизация недоступна в этой конфигурации. Информация, относящаяся к опорному сообщению, содержится в первом байте поля данных (8-бит). Формат поля данных опорного сообщения для режима Level1 приведён в следующей таблице:

Таблица 37.95. Формат поля данных опорного сообщения для режима Level1

Номер разряда	Условное обозначение	Назначение
7	NIG	Запрос паузы (GAP).
6	-	Резерв
5:0	CC	Номер цикла.

37.12.12.2 Условия перехода time master - time slave

После окончания инициализации контроллер, являющийся потенциальным time master, начинает отправлять опорные сообщения. Контроллер являющийся, time slave (TTOCF.OM=0), ожидает появления опорных сообщений. Если на шине несколько контроллеров в режиме потенциального time master, необходимо чтобы время отправки опорного сообщения каждого контроллера соответствовало приоритету указанному в поле TTRMC.RID[2:0]. Т. е. сначала опорное сообщение отправляет контроллер с самым высоким приоритетом, контроллер самым низким приоритетом должен передавать опорное сообщение последним. Если при передаче опорного сообщения контроллером с наивысшим приоритетом сбоя не произошло, то остальные контроллеры(в режиме потенциальный time master) не передадут опорные сообщения, так как цикл передачи начнётся сначала, а контроллер отправивший опорное сообщение последним будет текущим time master. Диаграмма переходов состояний time slave — time master приведена в следующем рисунке:

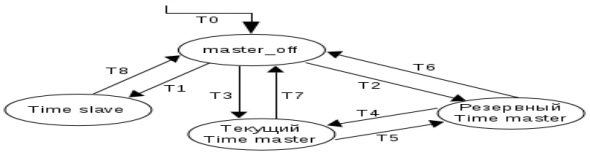


Рисунок 37.43. Диаграмма перехода состояний time master - time slave

Переход T0: аппаратный сброс, инициализация или конфигурационная ошибка (TTOST.EL=”11”);

Переход T1: принято опорное сообщение, но узел не является потенциальным time master;

Переход T2: принято опорное сообщение, с приоритетом ≠ собственному;

Переход T3: принято опорное сообщение, с приоритетом = собственному;

Переход T4: принято опорное сообщение, с приоритетом = собственному;

Переход T5: принято опорное сообщение, с более высоким приоритетом;

Переходы T6-T8: произошла конфигурационная ошибка (TTOST.EL=”11”).

37.12.12.3 Конфигурация NTU

NTU является мерой локального времени. Локальное время (16-битное значение) инкриминируется на каждый счёт NTU. Время цикла и глобальное время являются

производными от локального времени. Значения локального времени, глобального времени и времени цикла доступны в регистрах TTLGT и TTCTC.

В режиме Level2 длительность NTU определяется значением TUR (Time Unit Ratio). Значение TUR определяется формулой TUR = TURNA.NAV/TURCF.DC. Размер NTU задаётся формулой NTU = HCLK • TUR.

Значение TURNA.NAV определяется значением поля TURCF.NC. Поле NC это 18-битное значение, где TURCF.NCL [15:0] может быть запрограммировано в диапазоне 0x0000-0xFFFF, а TURCF.NCH [17:16] = 0x1. Когда в поле TURCF.NCL [15:0] записывается значение 0xnnnn, поле TURNA.NAV будет содержать значение 0x10000 + 0x0nnnn = 0x1nnnn. Значение делителя TURCF.DC это 14-битное значение. Диапазон TURCF.DC 0x0001 – 0x3FFF, 0x0000 – недопустимое значение.

Аппаратный сброс устанавливает TURCF.DC в 0x1000 и TURCF.NCL в 0x10000, в результате чего NTU состоит из 16 тактовых периодов. Локальное время и таймер watchdog не стартуют пока бит CCCR.INIT не сброшен, и TURCF.ELT не установлен. Установка TURCF.ELT в «1» блокирует запись в регистр TURCF.

При запуске, поле TURNA.NAV обновляется значением NC (= TURCF.NCL + 0x10000), когда TURCF.ELT = 1.

В режиме Level1 нет компенсации смещения. TURNA.NAV всегда равно NC.

В режиме Level2 есть две возможности изменения значения TURNA.NAV:

Когда работает как time slave или резервный time master, и когда установлен TTOCF.ECC. В этом случае TURNA.NAV автоматически корректируется исходя их значения глобального времени, при этом контроллер должен находиться в режиме синхронизации (In_Schedule) или паузы (In_Gap). Если синхронизация теряется, значение возвращается к NC.

Когда узел является текущим time master, и флаг TTOCF.EECS установлен, поле TURCF.NCL доступно для изменения программно. Для записи нового значения NC в поле TURNA.NAV необходимо установить TTOCN.ECS = 1, и поле TURNA.NAV обновится в момент начала следующего цикла (с отправкой следующего опорного сообщения). Статусный флаг TTOST.WECS устанавливается, когда TTOCN.ECS установлен и снимается после обновления TURNA.NAV. Запись в TURCF.NCL запрещена пока TTOST.WECS установлен.

В режиме Level2 процесс калибровки NTU подстраивает TURNA.NAV в диапазоне границы девиации синхронизации (SDL) NC ± 2^(TTOCF.LDSDL+5).

Девиация синхронизации SD это разница между TURCF.NC и TURNA.NAV (SD = |TURCF.NC - TURNA.NAV|). Ограничение девиации синхронизации SDL, задаётся

значением TTOCF.LDSDL (SDL = 2^ (TTOCF.LDSDL+5)) и должно быть мало относительно битового времени CAN шины. SD вычисляется при каждом новом цикле. Если TURNA.NAV отклоняется более чем на SDL от NC - устанавливается флаг TTIR.GTE, при этом TTOST.QCS сбрасывается. * В данной реализации GTE не используются *

37.12.12.4 Элемент события

До 64 элементов событий может использоваться для управления передачей сообщений во времени. Формат элементов событий описан в Разделе 37.10.6.

Поле метки времени (TM) описывает в какой момент цикла событие будет выполнено. Элементы событий должны храниться в порядке возрастания значения меток времени (поле TM). Поля MNR и CC игнорируются в элементах событий типа Tx_Ref_Trigger, Tx_Ref_Trigger_Gap, Watch_Trigger, Watch_Trigger_Gap и End_of_List.

Когда время цикла достигает значения метки времени текущего элемента события, из памяти читается следующий элемент события. Если элемент события должен передать сообщение, оно тоже читается из памяти. Необходимо, чтобы сообщение для передачи было прочитано до того, как время цикла достигнет метки времени элемента события. Так же необходимо чтобы сам элемент события был прочитан до того, как его метка времени будет достигнута. Если опорное сообщение имеет размер - n NTU, событие с меткой - TM <n никогда не станет активным и вероятно приведёт к конфигурационной ошибке.

Действия, производимые контроллером, работающим в режиме планирования событий, определяются типом элементов событий, которые выполняются планировщиком в соответствующие моменты времени:

Tx_Ref_Trigger (значение поля TYPE = “0000”);

Tx_Ref_Trigger_Gap (значение поля TYPE = “0001”);

Tx_Trigger_Single (значение поля TYPE = “0010”);

Tx_Trigger_Continous (значение поля TYPE = “0011”);

Watch_Trigger (значение поля TYPE = “0110”);

Watch_Trigger_Gap (значение поля TYPE = “0111”);

End_of_List (значение поля TYPE = “1000…1111”).

В начале каждого цикла, т.е. при каждом приёме или передаче опорного сообщения, список элементов событий обрабатывается начиная с первого элемента. Планировщик ищет первый элемент события, поле CC которого совпадает с номером текущего цикла. Планировщик ожидает пока время цикла достигнет метки времени элемента событие и

активирует его. Затем планировщик обращается к следующему элемент в списке, поле CC которого совпадает с текущим номером цикла и т.д.

Если узел является time master, и планировщик обнаруживает элемент события типа Tx_Ref_Trigger или Tx_Ref_Trigger_Gap запускается передача опорного сообщения. В случае, когда time slave обнаруживает Tx_Ref_Trigger(_Gap), это приводит к конфигурационной ошибке (TTOST.EL = “11”). Tx_Ref_Trigger_Gap используется только в режиме внешней синхронизации (TTOCF.OM = “10”). В состоянии синхронизации In_Gap (TTOST.SYS = “10”) Tx_Ref_Trigger игнорируется.

Событие Tx_Trigger_Single может использоваться для однократной передачи сообщения, для которого установлен бит регистра TXBRP. После успешной передачи бит сбрасывается.

Событие Tx_Trigger_Continous начинает эксклюзивную передачу сообщения, для которого установлен бит регистра TXBRP. После успешной передачи остаётся установленным и сообщение передаётся снова в следующем окне.

Допускается что несколько элементов событий передачи могут относится к одному передаваемому буферу. В зависимости от значения поля CC, элементы событий могут быть проигнорированы в некоторых циклах.

События Watch_Trigger (TYPE = “0110”) и Watch_Trigger_Gap (TYPE = “0111”) начинают проверку наличия пропущенных опорных сообщений. Они используются как в режимах time master так и time slave. Watch_Trigger_Gap используется только в режиме внешней синхронизации, в этом режиме Watch_Trigger игнорируется в состоянии синхронизации In_Gap (TTOST.SYS = “10”).

End_of_List (TYPE = “1000…1111”) не корректный тип события, если планировщик обнаруживает элемент события типа End_of_List перед Watch_Trigger или Watch_Trigger_Gap возникает ошибка конфигурации (TTOST.EL = “11”).

Конфигурация списка элементов событий:

Для обеспечения бесперебойного обмена, на элементы событий накладывается ряд ограничений:

Не должно быть двух элементов в одном цикле и с одинаковыми метками времени, однако события, которые выполняются в разных циклах (разные поля CC) могут иметь одинаковые метки времени.

События, которые помещаются после Watch_Trigger (Watch_Trigger_Gap) никогда не будут активны. В свою очередь событие Watch_Trigger никогда не выполнится если опорное сообщение принято вовремя.

Для всех не используемых элементов событий после Watch_Trigger (Watch_Trigger_Gap) необходимо установить тип End_of_List.

Особое внимание необходимо уделить временным меткам Tx_Ref_Trigger и Tx_Ref_Trigger_Gap. В резервном time master время запуска Tx_Ref_Trigger или Tx_Ref_Trigger_Gap складывается из значения его временной метки и значения смещения в поле TTOCF.IRTO. Значения временных меток других событий не должны находиться в этом диапазоне, в противном случае может получиться что временные метки появляются в неправильном порядке и возникнет конфигурационная ошибка.

Конфигурационная ошибка, отражаемая в поле TTOST.EL = “11”, возникает в случаях:

Поле CC (код цикла) элемента события соответствует текущему номеру цикла, а значение поля TM (метка времени) меньше, чем текущее время цикла.

Контроллер в режиме time slave (TTOCF.TM = “0”) обнаруживает событие с типом Tx_Ref_Trigger или Tx_Ref_Trigger_Gap.

Контроллер обнаруживает что, значение поля TM (метка времени) следующего события находится в пределах окна Tx enable (TTMLM.TXEW) текущего элемента события передачи, и у них совпадают поля CC (код цикла).

Значение полей TM соседних элементов событий, первый из которых имеет тип Tx_Ref_Trigger отличаются на величину меньшую, чем значение TTOST.RTO.

Пример формирования системной матрицы на основе элементов событий:

Возможная последовательность элементов событий в памяти представлена в следующей таблице:

Таблица 37.96. Пример списка элементов событий

событие	Метка времени TM[15:0]	код цикла CC[6:0]	Тип элемента события Trigger TYPE[3:0]	Номер сообщения MNR[6:0]
0	Метка1	7'b0000100	Tx_Trigger_Single	7
1	Метка1	7'b1000000	Tx_Trigger_Single	3
2	Метка1	7'b1000011	Tx_Trigger_Single	7
3	Метка3	7'b1000001	Tx_Trigger_Single	2
4	Метка3	7'b1000011	Tx_Trigger_Single	5
5	Метка4	-	Tx_Ref_Trigger	-
6	Метка5	-	Watch_Trigger	-
7	Метка6	-	Tx_Ref_Trigger_Gap	-
8	Метка7	-	Watch_Trigger_Gap	-
9	-	-	End_of_List	-

Отсчет циклов начинается с номера 0 и продолжается до значения, указанного в поле TTMLM.CCM (поскольку отсчет циклов начинается с 0, значение CCM = желаемое количество циклов — 1). Затем матричный цикл повторяется снова. Поле код цикла (CC) элемента события содержит два параметра:

разряд, в котором содержится старшая '1' — повторяемость выполнения события;

остальные разряды справа от старшей '1' — номер цикла с которого начнётся выполнение события.

Например: CC = 7'b0010011 — повторение через каждые 16 циклов начиная с 3 цикла.

Поле метка времени (TM) определяет в какой момент времени цикла будет выполнено событие. Тип события (TYPE) определяет действие которое будет выполнено в момент, когда это событие станет активным. Если при выполнении события должна произойти передача сообщения то поле MNR определяет из какого элемента буфера Tx Buff (0..31) будет взято сообщение. Если при выполнении элемента события ожидается принятое сообщение, то поле MNR содержит номер фильтра, которому соответствовало принятое сообщение. В зависимости от значения FTYPE значение MNR будет относится либо к фильтру стандартных сообщений, либо к фильтру расширенных.

На основании списка элементов событий из Таблица 37.96 и значения TTMLM.CCM = 0x4 (5 циклов) можно составить системную матрицу:

Таблица 37.97. Системная матрица

Номер цикла CC	Метка1	Метка2	Метка3	Метка4	Метка5	Метка6	Метка7
0	Tx7* (событие 0)	-	-	TxRef	WatchTrigger	TxRefGap	WatchTriggerGap
1	Tx3 (событие 1)	-	Tx2 (событие 3)	TxRef	WatchTrigger	TxRefGap	WatchTriggerGap
2	-	-	-	TxRef	WatchTrigger	TxRefGap	WatchTriggerGap
3	Tx7 (событие 2)	-	Tx5 (событие 4)	TxRef	WatchTrigger	TxRefGap	WatchTriggerGap
4	Tx7 (событие 0)	-	-	TxRef	WatchTrigger	TxRefGap	WatchTriggerGap

*Txn — передача сообщения из буфера n.

Для элементов событий типа Tx_Ref_Trigger поле CC игнорируется, они выполняются в каждом цикле.

37.12.12.5 Инициализация режима планирования событий

Синхронизация работы контроллера начинается после сброса CCCR.INIT. Контроллер может работать как в режиме планирования событий (TTOCF.GEN = “0”) так и в режиме планирования событий с внешней синхронизацией (TTOCF.GEN = “1”). Все узлы начинают работу со значением времени цикла = 0 и режимом синхронизации TTOST.SYS = “00” (не синхронизировано), все передачи запрещены, кроме передачи опорных сообщений. Узлы в режиме планирования с внешней синхронизацией будут игнорировать события Tx_Ref_Trigger и Watch_Trigger и вместо этого необходимо использовать Tx_Ref_Trigger_Gap и Watch_Trigger_Gap.

После конфигурации, контроллер в режиме time slave будет игнорировать события Watch_Trigger и Watch_Trigger_Gap до получения опорного сообщения.

Если контролер не принял опорное сообщение до активизации Watch_Triggers, это приведёт к конфигурационной ошибке (TTOST.EL = “11”), контроллер выставит флаг прерывания TTIR.WT и перейдёт в режим монитора (CCCR.MON = “1”). В режиме монитора контроллер все ещё может принимать сообщения, но не может отправлять доминантные биты, а, следовательно, не может отправить флаг acknowledge.

Если в процессе синхронизации ошибок не обнаружено, первое опорное сообщение меняет статус синхронизации на TTOST.SYS = “01” (Синхронизация), второе (в зависимости от бита Next_is_Gap) на TTOST.SYS = “11” (In_Schedule) или TTOST.SYS = “10” (In_Gap), разрешая выполнение событий.

После конфигурации, контроллер в режиме потенциального time master передаст опорное сообщение, когда время цикла достигнет метки времени Tx_Ref_Trigger (Tx_Ref_Trigger_Gap). Элементы событий Watch_Trigger и Watch_Trigger_Gap будут игнорироваться пока контроллер не получит какое-либо сообщение или не передаст опорное сообщение.

Потенциальный time master станет текущим time master (TTOST.MS = “11”) если он передал последнее опорное сообщение в цикле, в противном случае он будет резервным time master (TTOST.MS = “10”).

37.12.12.6 Внешняя синхронизация в режиме планирования событий

В режиме планирования с внешней синхронизацией (TTOCF.GEN = “1”) отправка сообщений может прерываться паузами (Gaps) между циклами системной матрицы. При этом все узлы подключённые к CAN сети должны работать в режиме планирования с внешней синхронизацией.

Во время паузы (Gap), все передачи останавливаются, и шина CAN находится в состоянии ожидания. Пауза (Gap) заканчивается, когда следующее опорное сообщение начинает новый цикл. Пауза (Gap) начитается в конце цикла, который был начат опорным сообщением с установленным битом Next_is_Gap = “1”. Паузы (Gaps) инициируются текущим time master.

Текущий time master может инициировать паузу (Gap) двумя способами:

Программно, записью TTOCN.NIG = “1”. Бит Next_is_Gap будет передан как “1” в следующем опорном сообщении.

Аппаратно, если установлен TTOCN.GCS = “1”, тогда сигнал на внешнем входе EXT_EVENT в момент отправки опорного сообщения инициирует паузу в конце цикла. При

этом бит Next_is_Gap = “1” содержится в каждом опорном сообщении текущего time master, соответственно, time slave постоянно ожидает паузы.

Как только опорное сообщение отправлено, устанавливается бит TTOST.WFE, сообщая о предстоящей паузе (Gap) для time master и time slave. Пауза наступает по окончании последнего временного окна.

Для всех тайм мастеров, бит TTOST.GSI будет установлен, когда последний цикл закончится и начнётся пауза (Gap). Для всех узлов в режиме time slave, бит TTOST.GSI останется “0”.

Когда потенциальный time master находится в состоянии In_Gap (TTOST.SYS = “10”), есть 4 способа выйти из этого состояния:

Программно, записью TTOCN.FGP = “1”

Аппаратно, по заднему фронту сигнала на входе EXT_EVENT. Задний фронт на входе EXT_EVENT инициирует отправку опорного сообщения и окончание паузы.

Любой потенциальный тайм мастер прервёт «паузу» (Gap), когда достигнет метки времени своего Tx_Ref_Trigger_Gap элемента, означающее что синхро-событие не произошло вовремя.

Установка TTOCN.FGP после начала паузы немедленно инициирует передачу опорного сообщения тем самым синхронизируя планировщик. Если TTOCN.FGP установлен до начала паузы (во время цикла) пауза не произойдёт.

В режиме планирования событий без внешней синхронизации, бит Next_is_Gap = “1” в опорном сообщении будет проигнорирован, как и сигнал на входе EXT_EVENT и биты TTOCN.NIG, TTOCN.FGP.

37.12.12.7 Локальное время, Время цикла, Глобальное время и внешняя синхронизация

Локальное время — это 16-битный циклический счётчик, увеличивающийся с каждым шагом NTU.

Любое сообщение принятое или переданное приводит к захвату локального в момент начала кадра (SoF), в этот момент значение локального времени сохраняется как Sync_Mark.

При приеме или передаче опорного сообщения, внутренние Ref_Mark обновляются значением Sync_Mark. Разница между Ref_Mark и текущим значением локального времени называется временем цикла (Время цикла = локальное время – Ref_Mark) и доступно в TTCTC.CT.

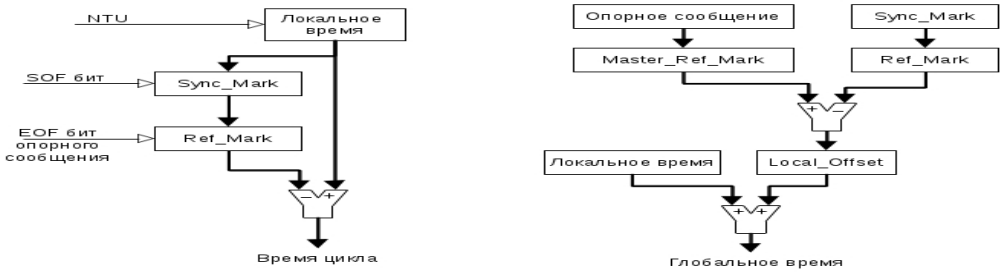


Рисунок 37.44. Синхронизация времени цикла и глобального времени

Начало отсчёта времени цикла это первый бит (SOF) опорного сообщения. Следующее опорное сообщение запрашивается, когда время цикла достигает метки времени TM элемента события Tx_Ref_Trigger. Новое значение Sync_Mark захватывается на SOF бите передаваемого сообщения, но время цикла будет увеличиваться пока опорное сообщение не будет передано (или принято) и Sync_Mark будет воспринят как новый Ref_Mark. В этот момент, время цикла перезапустится. Как следствие, время цикла может никогда (кроме начального запуска) не принимать значение <n, где n это длина опорного сообщения выраженная в NTU.

Пример: На Рисунок 37.45 схематично изображён момент начала нового цикла в момент появления опорного сообщения. Отсчёт времени цикла начинается с 6, т.к. с момента захвата значения Sync Mark до присваивания этого значения Ref Mark проходит время в 6(NTU), равное длительности опорного сообщения (значение будет отличаться от рабочего и приведено только для наглядности).

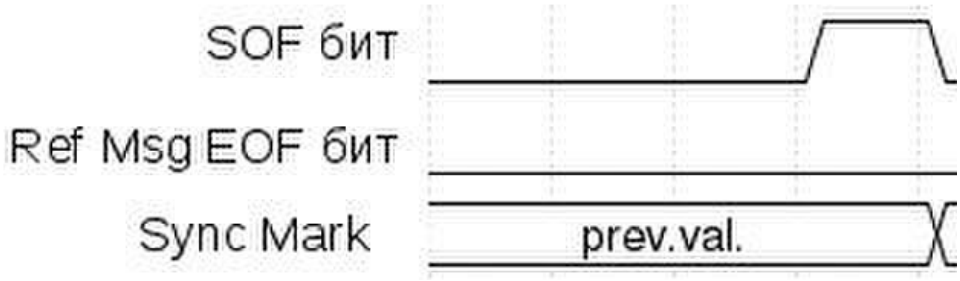


Рисунок 37.45. Смещение начала отсчета времени цикла

Глобальное время поддерживается только в режиме Level2. После конфигурации, потенциальный time master будет транслировать свое локальное время, как глобальное время, для других узлов. Time master передаёт свои Ref_Marks как Master_Ref_Marks в опорном сообщении, прочитанное из регистра TTLGT.GT. Глобальное время это сумма локального времени узла и его локальное смещение.

Узел, принявший опорное сообщение, вычисляет локальное смещение относительно глобального времени сравнивая свой Ref_Mark с принятым Master_Ref_Mark. Глобальное время в представлении отдельного узла это локальное время+ локальное смещение. Потенциальный time master, который никогда не принимал опорного сообщения, будет

иметь нулевое локальное смещение. Когда узел станет текущим time master, после приёма опорного сообщения, локальное смещение будет зафиксировано своим последним значением.

Контроллер сравнивает длительность цикла в масштабе локального и глобального времени. Если между этими значениями есть разница – будет вычислено новое значение TURNA.NAV. Если размер девиации синхронизации не превышен SD = |NC - TURNA.NAV | ≤ SDL (Synchronisation Deviation Limit), значение TURNA.NAV будет обновлено. В противном случае автоматическая компенсация смещения будет отложена.

Корректировка рассчитывается по формуле TURтекущий = TURпредыдущий*df,

Где df = (Ref_Marktек - Ref_Markпред)/(Master_Ref_Marktек - Master_Ref_Markпред)

В режиме Level2, статус автоматической компенсации смещения отражает бит TTOST.QCS. В режиме Level1, TTOST.QCS всегда «1».

TTOST.QGTP показывает совпадают ли собственное глобальное время узла и глобальное время узла, являющегося текущим time master. TTOST.QGTP всегда в «0» в режиме Level1, и когда превышен предел девиации синхронизации в режиме Level2 (TTOST.QCS = “0”).

37.12.12.8 Передача сообщений в режиме планирования событий

Передача осуществляется из буферов Tx Buff. Поле MNR элемента события указывает на элемент буфера Tx Buff содержащий сообщение, которое необходимо передать. Если сообщение предназначено для передачи более одного раза за цикл, поле MNR нескольких событий будет совпадать.

Данные должны обновляться регулярно и вовремя. Для избежания отправки не полностью обновленных сообщений рекомендуется:

для событий Tx_Trigger_Single:

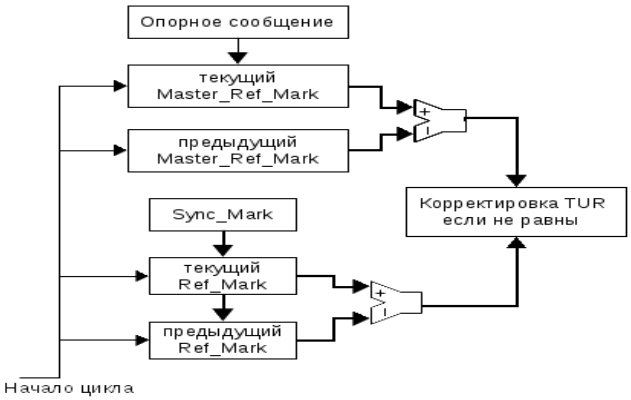


Рисунок 37.46. Схема компенсации смещенного времени

Проверить завершена ли предыдущая передача чтением TXBTO;

Обновить буфер передачи;

Сделать запрос на передачу TXBAR соответствующим битом.

Для событий Tx_Trigger_Continous:

Сделать запрос на отмену передачи TXBCR сбросив бит в TXBRP;

Проверить завершена ли отмена передачи чтением TXBCF;

Обновить буфер передачи;

Сделать запрос на передачу TXBAR соответствующим битом.

Если буфер передачи, имеющий активный запрос на передачу, должен быть обновлен, необходимо сначала выставить запрос на отмену и проверить была ли она завершена, прочитав TXBCF перед началом обновления.

Передача происходит, когда время цикла достигает метки времени события Tx_Trigger_Single или Tx_Trigger_Continous. После успешной передачи в результате события Tx_Trigger_Single соответствующий бит TXBRP сбрасывается. После успешной передачи в результате события Tx_Trigger_Continous соответствующий бит TXBRP остаётся установлен. Если передача окончилась неудачей из-за помех, она повторится в следующий раз, когда одно из событий станет активным.

37.12.13 Прерывания

Для работы с прерываниями в контроллере CAN предусмотрены регистры IR, IE, TTIR, TTIE.

Регистр IR – основной регистр прерываний.

Регистр IE – регистр разрешения прерываний в IR.

Регистр TTIR – регистр прерываний для режима планирования.

Регистр TTIE – регистр разрешения прерываний в TTIR.

Разряды регистра IR объединяются по «ИЛИ» и определяют значения выхода прерываний INT0.

Разряды регистра TTIR объединяются по «ИЛИ» и определяют значения выхода прерываний INT1.

38. КОНТРОЛЛЕР WATCHDOG TIMER (WDT)

38.1 Общая информация

Сторожевой таймер WDT служит для предотвращения зависаний системы. Таймер постоянно декрементируется от исходного состояния до нуля. Если таймер программно не вернуть в исходное положение до достижения нуля, то блок сгенерирует системный сброс или прерывание.

Микросхема содержит 2 контроллера wdt в подсистемах service_subs и lsperiph1_subs. Отличия таймеров приведены в Таблица 38.1

Таблица 38.1 Отличия параметров WDT

Параметр	WDT0 (service_subs)	WDT1 (lsperiph1_subs)
Источник опорной частоты таймера	service_subs ucg1/clk_o[0]	lsperiph1_subs ucg/clk_o[9]
Генерация прерывания и системного сброса	Генерирует прерывание и системный сброс	Генерирует прерывание.

38.2 Регистры WDT

Перечень регистров WDT приведен в Таблица 38.2.

Таблица 38.2. Перечень программно-доступных регистров WDT

Условное обозначение	Название регистра	Тип доступа	Исходное состояние wdt0	Исходное состояние wdt1	Смещение
WDT_CR	Регистр управления таймером	W/R	0xA	0x2	0x0
WDT_TORR	Регистр диапазона таймаута	W/R	0xFF	0x0	0x4
WDT_CCVR	Регистр текущего значения счетчика	R	0x7FFFFFFF	0xFFFF	0x8
WDT_CRR	Регистр сброса счетчика	W	0x0	0x0	0xC
WDT_STAT	Регистр статуса прерывания	R	0x0	0x0	0x10
WDT_EOI	Регистр сброса прерывания	R	0x0	0x0	0x14
WDT_COMP_PARAM_5	Регистр 5 параметров контроллера	R	0x0	0x7FFFFFFF	0xE4
WDT_COMP_PARAM_4	Регистр 4 параметров контроллера	R	0x0	0x7FFFFFFF	0xE8

WDT_COMP_PARAM_3	Регистр 3 параметров контроллера	R	0xFF	0x0	0xEC
WDT_COMP_PARAM_2	Регистр 2 параметров контроллера	R	0x7FFFFFFF	0xFFFF	0xF0
WDT_COMP_PARAM_1	Регистр 1 параметров контроллера	R	0x10FF0A46	0x10000286	0xF4
WDT_COMP_VERSION	Регистра версии контроллера	R	0x3131302A	0x3131302A	0xF8
WDT_COMP_TYPE	Регистр типа контроллера	R	0x44570120	0x44570129	0xFC

38.2.1 Регистр WDT_CR

Регистр управления WDT.

Формат регистра WDT_CR приведен в Таблица 38.3.

Таблица 38.3. Формат регистра WDT_CR

Номер бита	Условное обозначение	Назначение
0	WDT_EN	Бит включения WDT. Пока WDT не включен, его счетчик не декрементируется, а значит не генерирует прерывание и системный сброс. Будучи однажды включённым, WDT выключается только после генерации системного сброса.
1	RMOD	Включение прерывания. 0: после таймаута генерируется системный сброс. 1: после таймаута генерируется прерывание. Если к следующему таймауту прерывание не будет сброшено, то генерируется системный сброс. В контроллере wdt1 этот бит включает/выключает генерацию прерывания. Системный сброс от wdt1 не будет сгенерирован в любом случае.
2:4	RPL	Продолжительность импульса системного сброса в тактах опорной частоты: 000 – 2 такта 001 – 4 такта 010 – 8 тактов 011 – 16 тактов 100 – 32 такта 101 – 64 такта 110 – 128 тактов 111 – 256 тактов
5:31	-	Резерв

38.2.2 Регистр WDT_TORR

Регистр таймаута WDT.

Формат регистра WDT_TORR приведен в Таблица 38.4.

Таблица 38.4. Формат регистра WDT_TORR

Номер бита	Условное обозначение	Назначение
0:3	TOP	Период таймаута. Поле хранит значение, с которого начинает отсчёт таймер. Изменение этого поля будет иметь эффект только после сброса таймера. Период таймаута рассчитывается следующим образом. T = 2(16 + i), где T – период, выраженный в тактах опорной частоты, а i – значение поля TOP.
4:7	TOP_INT	Период таймаута для инициализации. Значение этого поля будет записано в TOP после включения таймера. Запись в регистр возможна после системного сброса, но до включения WDT.
8:31	-	Резерв

38.2.3 Регистр WDT_CCVR

Формат регистра WDT_CCVR приведен в Таблица 38.5.

Таблица 38.5. Формат регистра WDT_CCVR

Номер бита	Условное обозначение	Назначение
0:31	CCVR	Регистр содержит текущее значение счетчика WDT.

38.2.4 Регистр WDT_CRR

Регистр сброса счетчика WDT.

Формат регистра WDT_CRR приведен в Таблица 38.6.

Таблица 38.6. Формат регистра WDT_CRR

Номер бита	Условное обозначение	Назначение
0:7	CRR	Для перезапуска счётчика необходимо записать 0x76 в это поле. Также, это сбросит прерывание от WDT.
8:31	-	Резерв

38.2.5 Регистр WDT_STAT

Регистр статуса прерывания WDT.

Формат регистра WDT_STAT приведен в Таблица 38.7.

Таблица 38.7. Формат регистра WDT_STAT

Номер бита	Условное обозначение	Назначение
0	ISR	Статус прерывания WDT: 1 – прерывание активно. 0 – прерывания не активно
1 :31	-	Резерв

38.2.6 Регистр WDT_EOI

Регистр сброса прерывания.

Формат регистра WDT_EOI приведен в Таблица 38.8.

Таблица 38.8. Формат регистра WDT_EOI

Номер бита	Условное обозначение	Назначение
0	ICR	Чтение этого регистра сбрасывает прерывание WDT.
1:31	-	Резерв

38.3 Функционирование

После включения WDT, счетчик начинает отсчет от предустановленного значения до нуля. При достижении нуля, в зависимости от выбранного режима, генерируется либо системный сброс, либо прерывание. Режим выбирается полем RMOD регистра WDT_CR. При достижении нуля в счетчик заново загружается значение таймаута, и он продолжает декрементироваться. Пользователь в любой момент может сбросить счетчик на исходное состояние записью 0x76 в регистр WDT_STAT.

В случае, если RMOD=1, WDT установил прерывание, и оно не было сброшено до следующего достижения счетчиком нуля, то WDT генерирует системный сброс. Прерывание сбрасывается чтением регистра WDT_EOI или записью 0x76 в регистр WDT_STAT.

39. БЛОК УНИВЕРСАЛЬНЫХ ТАЙМЕРОВ (TIMERS)

39.1 Общая информация

В микросхеме интегрированы два блока универсальных таймеров:

1. Timer0 находится в подсистеме низкоскоростных блоков lsperiph1, и содержит восемь независимых каналов (N=8) два из которых (N=1 и N=2) способны генерировать сигнал импульсной модуляции;

2. Timer1 находится в подсистеме SDR, и содержит четыре независимых канала (N=4)

При включении таймеры начинают обратный отсчет от установленного значения и выдают прерывание по достижению нуля.

39.2 Регистры

В Таблица 39.1 приведен перечень программно-доступных регистров блока универсальных таймеров.

Таблица 39.1. Перечень программно-доступных регистров блока универсальных таймеров

Условное обозначение регистра	Название регистра	Тип Доступа	Исходное Состояние	Смещение
Timer*N*LoadCount	Значение счетчика для загрузки в Timer*N*.	W/R	0x0	N=1: 0x00 N=2: 0x14 N=3: 0x28 N=4: 0x3C N=5: 0x50 N=6: 0x64 N=7: 0x78 N=8: 0x8C
Timer*N*CurrentValue	Регистр текущего значения Timer*N*.	R	0x0	N=1: 0x04 N=2: 0x18 N=3: 0x2C N=4: 0x40 N=5: 0x54 N=6: 0x68 N=7: 0x7C N=8: 0x90
Timer*N*ControlReg	Регистр управления Timer*N*.	W/R	0x0	N=1: 0x08 N=2: 0x1C N=3: 0x30 N=4: 0x44 N=5: 0x58 N=6: 0x6C N=7: 0x80 N=8: 0x94
Timer*N*EOI	Регистр сброса прерывания Timer*N*.	R	0x0	N=1: 0x0C N=2: 0x20 N=3: 0x34

Условное обозначение регистра	Название регистра	Тип Доступа	Исходное Состояние	Смещение
				N=4: 0x48 N=5: 0x5C N=6: 0x70 N=7: 0x84 N=8: 0x98
Timer*N*IntStatus	Регистр статуса прерывания Timer*N*.	R	0x0	N=1: 0x10 N=2: 0x24 N=3: 0x38 N=4: 0x4C N=5: 0x60 N=6: 0x74 N=7: 0x88 N=8: 0x9C
TimersIntStatus	Регистр статуса прерываний всех.	R	0x0	0xA0
TimersEOI	Регистр сброса всех активных прерываний.	R	0x0	0xA4
TimersRawIntStatus	Регистр статуса всех немаскированных прерываний.	R	0x0	0xA8
TIMERS_COMP_VERSION	Регистр содержащий номер ревизии блока	R	0x 3231312a	0xAC
Timer*N*LoadCount2	Значение счетчика для загрузки в Timer*N* после переключения выхода таймера из 0 в 1.	W/R	0x0	N=1: 0xB0 N=2: 0xB4
TIMER_N_PROT_LEVEL	Регистр управления уровнем защиты доступа	W/R	0x0	N=1: 0xD0 N=2: 0xD4

39.2.1 Регистр Timer*N*LoadCount

Формат регистра Timer*N*LoadCount приведен в Таблица 39.2.

Таблица 39.2. Формат регистра Timer*N*LoadCount

Номер бита	Условное обозначение	Назначение
0:31	Timer*N*LoadCount	Значение этого регистра загружается в Timer*N* перед началом отсчета. Значение содержит точку начала отсчета Timer*N*.

39.2.2 Регистр Timer*N*CurrentValue

Формат регистра Timer*N*CurrentValue приведен в Таблица 39.3.

Таблица 39.3. Формат регистра Timer*N*CurrentValue

Номер бита	Условное обозначение	Назначение
0:31	Timer*N*CurrentValue	Текущее значение таймера *N*.

39.2.3 Регистр Timer*N*ControlReg

Этот регистр управляет включением/выключением, режимом работы и маскированием прерываний Timer*N*.

Формат регистра Timer*N*ControlReg приведен в Таблица 39.4.

Таблица 39.4. Формат регистра TimerNControlReg

Номер бита	Условное обозначение	Назначение
0	TIMER_ENABLE	Бит включения Timer*N*. 0: выключен. 1: включен.
1	TIMER_MODE	Бит режима работы Timer*N*. 0: free-running 1: определяемый пользователем.
2	TIMER_INTERRUPT_MASK	Бит маскирования прерывания от Timer*N*. 0: прерывание разрешено. 1: прерывание запрещено.
3	TIMER_PWM	Бит включения выходного сигнала импульсной модуляции timer_*N*_toggle. 0: Генерация сигнала ШИМ выключена. 1: Генерация сигнала ШИМ включена.
4	TIMER_0N100_PWM_EN	Бит включения режима 0% и 100% скважности режима ШИМ. 0: Режим выключен 1: Режим включен Поле доступно только для первого и второго счетчика блока Timers0 подсистемы lsperiph1
5:31	-	Резерв

39.2.4 Регистр Timer*N*EOI

Формат регистра Timer*N*EOI приведен в Таблица 39.5.

Таблица 39.5. Формат регистра Timer*N*EOI

Номер бита	Условное обозначение	Назначение
0	Timer*N*EOI	Чтение этого регистра возвращает 0 и сбрасывает прерывание таймера *N*.
1:31	-	Резерв

39.2.5 Регистр Timer*N*IntStatus

Формат регистра Timer*N*IntStatus приведен в Таблица 39.6.

Таблица 39.6. Формат регистра Timer*N*IntStatus

Номер бита	Условное Обозначение	Назначение
0	Timer*N* IntStatus	Регистр содержит статус прерывания таймера *N*.
1:31	-	Резерв

39.2.6 Регистр TimersIntStatus

Формат регистра TimersIntStatus приведен в Таблица 39.7.

Таблица 39.7. Формат регистра TimersIntStatus

Номер бита	Условное Обозначение	Назначение
0:7	TimersIntStatus	Регистр содержит статусы прерываний всех таймеров.
8:31	-	Резерв

39.2.7 Регистр TimersEOI

Формат регистра TimersEOI приведен в Таблица 39.8.

Таблица 39.8. Формат регистра TimersEOI

Номер бита	Условное Обозначение	Назначение
0:7	TTIMERSEOI	Чтение этого регистра возвращает нули и сбрасывает все активные прерывания.
8:31	-	Резерв

39.2.8 Регистр TimersRawIntStatus

Формат регистра TimersIntRawStatus приведен в Таблица 39.9.

Таблица 39.9. Формат регистра TimersRawIntStatus

Номер бита	Условное обозначение	Назначение
0:7	Timers Interrupt Status Register	Регистр содержит статусы прерываний всех незамаскированных таймеров.
8:31	-	Резерв

39.2.9 Регистр TIMERS_COMP_VERSION

Формат регистра TIMERS_COMP_VERSION приведен в Таблица 39.10.

Таблица 39.10. Формат регистра TIMERS_COMP_VERSION

Номер бита	Условное обозначение	Назначение
0:31	TIMERSCOMPVERSION	Регистр конфигурации блока.

39.2.10 Регистр TimerNLoadCount2

Формат регистра Timer*N*LoadCount2 приведен в Таблица 39.11.

Таблица 39.11. Формат регистра Timer*N*LoadCount2

Номер бита	Условное обозначение	Назначение
0:31	TIMERLOADCOUNT2	Это значение загружается в Timer*N* (N=1,2), когда Timer*N*ControlReg[3]=1 (N=1,2) и выходной сигнал timer_*N*_toggle изменяется с 0 на 1. Это значение определяет продолжительность высокого уровня сигнала timer_*N*_toggle.

39.2.11 Регистр TIMER_N_PROT_LEVEL

Формат регистра TIMER_N_PROT_LEVEL приведен в Таблица 39.12.

Таблица 39.12. Формат регистра TIMER_N_PROT_LEVEL

Номер бита	Условное обозначение	Назначение
0:2	Timer_N_ProtLevelField	Включение защиты доступа к регистрам управления счетчиками Доступ разрешен при совпадении значения на линиях системной шины PPROT [2:0] со значением в данном поле Поле доступно только для блока Timers0 подсистемы lsperiph1
3:31	-	Резерв

39.3 Функционирование

39.3.1 Включение и выключение таймера

Для включения или выключения Timer*N*, необходимо записать соответственно 1 или 0 в бит 0 регистра Timer*N*ControlReg.

Когда таймер включен, значение его счетчика декрементируется каждый такт частоты:

1. В подсистеме lsperiph1 частота инкремента формируется выделенном только для таймеров 7-м канале ucg.clk_o[7] и может быть выбрана программно
2. В подсистеме sdr таймеры тактируются общей системной частотой cfg_clk, которая формируется на 0-м канале ucg0.clk_o[0]

При включении таймера, текущее значение регистра Timer*N*LoadCount загружается в счетчик таймера.

При выключении таймера, значение его счетчика сбрасываются.

39.3.2 Загрузка значения начала отсчета

При включении таймера, значение счетчика загружается из регистра Timer*N*LoadCount, это происходит в обоих free-running и user-defined режимах.

Когда счетчик таймера достигает нулевого состояния, в счетчик загружается одно из двух значений, в зависимости от выбранного режима работы таймера.

В режиме user-defined - в таймер загружается текущее значение регистра Timer*N*LoadCount. Этот режим используется для генерирования периодических прерываний. Режим можно выбрать, записав 1 в бит 1 регистра Timer*N*ControlReg.

В режиме free-running - в таймер загружается максимально возможное значение 0xFFFFFFFF, что позволяет перепрограммировать или выключить таймер, до того, как произойдет новое прерывание. Этот режим используется для генерирования единичных прерываний. Этот режим можно выбрать, записав 0 в бит 1 регистра Timer*N*ControlReg.

39.3.3 Импульсная модуляция

Два из восьми таймеров блока универсальных таймеров, интегрированных в подсистеме lsperiph1, имеют выходной сигнал импульсной модуляции timer_*N*_toggle(N=1,2).

Продолжительность высокого и низкого уровня сигнала модуляции, выраженная в тактах частоты тактирования счетчика, определяется регистрами TimerNLoadCount2 и TimerNLoadCount:

HIGH_period = (Timer*N*LoadCount2 + 1)

LOW_period = (Timer*N*LoadCount + 1)

40. ТАЙМЕР РЕАЛЬНОГО ВРЕМЕНИ (RTC)

40.1 Основные характеристики

1. Подсчет лет, месяцев, дней недели, дней, часов, минут, секунд и шестнадцатых долей секунды.
2. Поддержка високосного года.
3. Представление данных в двоично-десятичном коде (BCD).
4. Будильника — генерирует прерывание по достижению заданного времени.
5. Интервальный таймер — генерирует периодическое прерывание с заданным интервалом.
6. Вывод системы из режима power down по сигналу от будильника или интервального таймера.
7. APB — интерфейс.
8. Наличие механизма контроля целостности данных в операциях чтения и записи.
9. Поддержка работы от автономного питания.
10. Функционирование на частоте 32,768 кГц.

40.2 Структурная схема

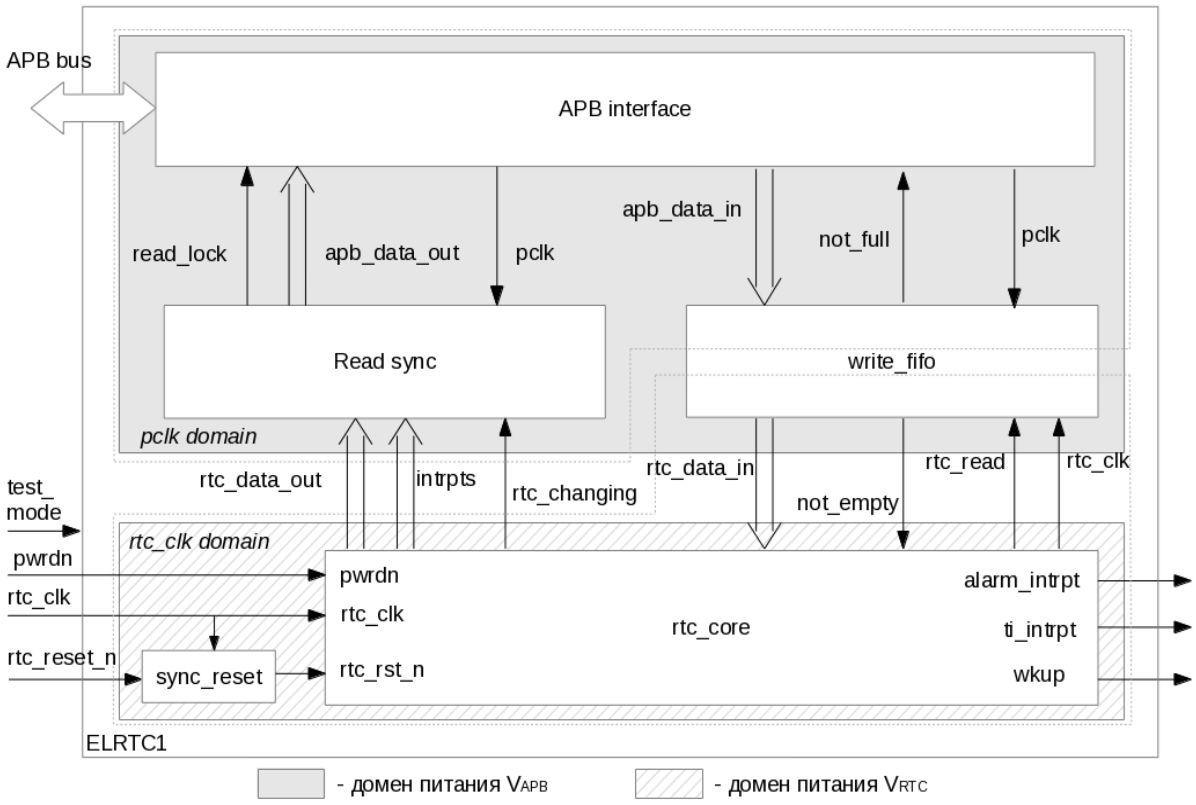


Рисунок 40.1. Структурная схема IP блока ELRTC1

IP блок ELRTC1 имеет два домена синхронизации: домен частоты APB интерфейса — *pclk*, и домен частоты *rtc_clk* равной 32.768 кГц. Интерфейс APB функционирует на частоте шины *pclk* и передает данные в rtc_core через двух портовый модуль *write_fifo* с указателями на счетчиках Грея. Размер FIFO, находящегося в этом модуле, составляет 8 слов по 38 бит. По мере появления данных на выходах *write_fifo*, *rtc_core* по положительному фронту *rtc_clk* сохраняет их во внутренних регистрах. Все внутренние регистры *rtc_core* доступны на чтение. Функционал Read Sync обеспечивает контроль целостности считывающихся данных.

40.3 Механизм контроля целостности данных в пределах регистра в операциях чтения

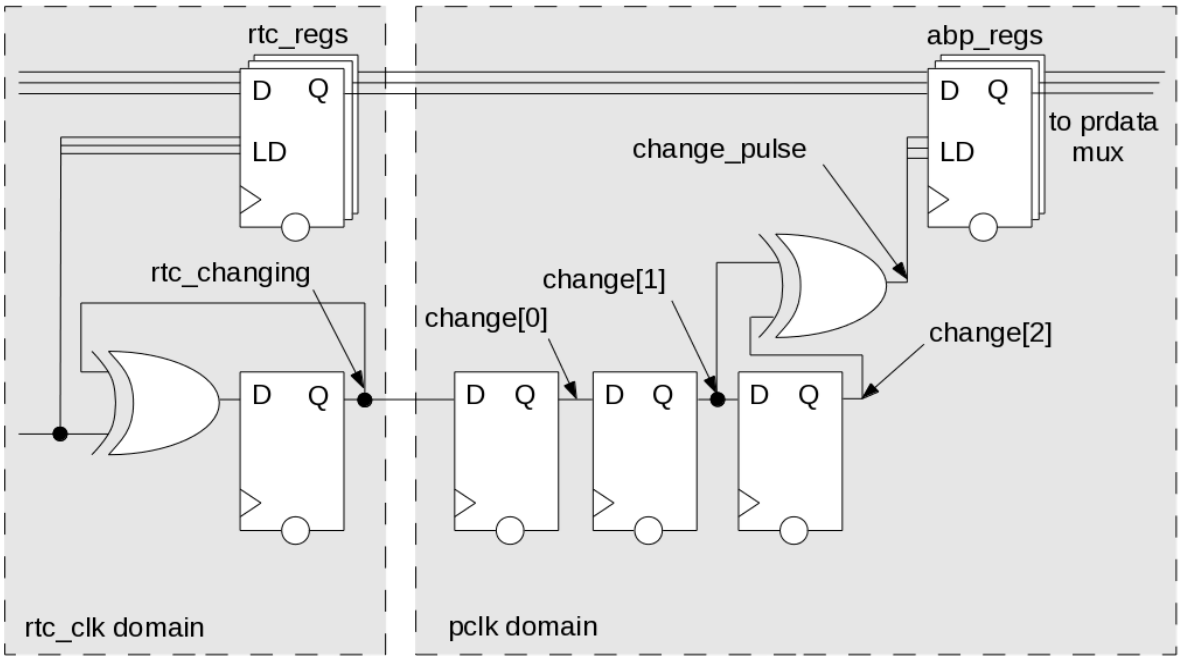


Рисунок 40.2. Логическая схема механизма синхронизации при чтении

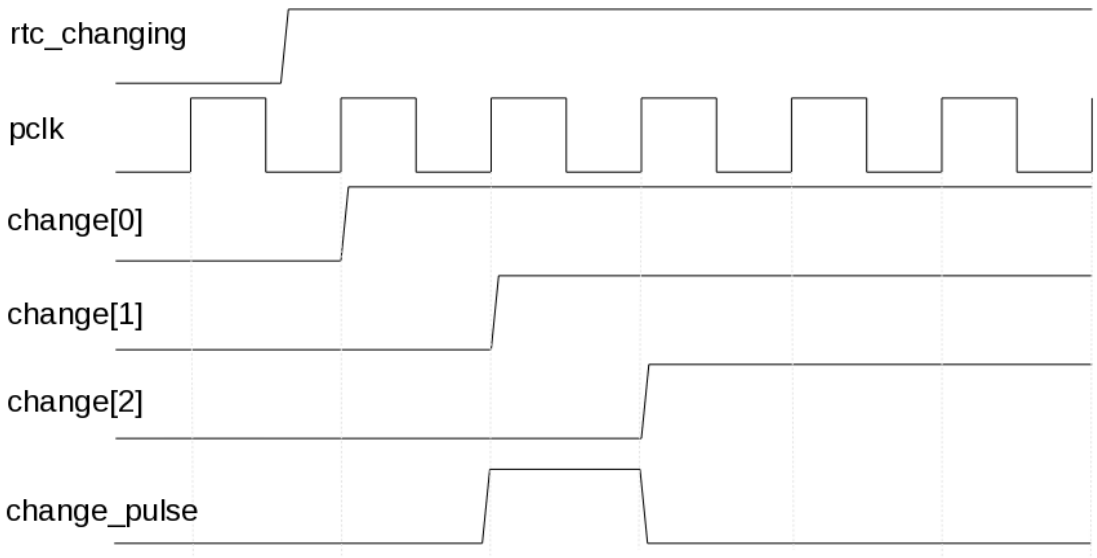


Рисунок 40.3. Временная диаграмма синхронизации при чтении

Необходимость контроля целостности данных во время операций записи возникает из-за ограниченности размера FIFO. Большая разница в частотах *pclk* и *rtc_clk*, может привести к тому, что FIFO переполнится и часть данных с системной шины не будут записаны. Для исключения подобной ситуации во время частой записи программное обеспечение должно периодически проверять количество занятых ячеек памяти в буфере FIFO. Реализуется это

чтением поля USED_CELLS регистра RRTC_CTRL. При полном заполнении буфера, далее записываемые данные будут игнорироваться.

Целостность данных во время операций чтения также контролируется. На рисунке 13.2 представлена схема синхронизации передачи данных из частотного домена *rtc_clk* в домен повышенной частоты *pclk*. Каждое изменение регистров в частотном домене rtc_clk сопровождается изменением фронта сигнала rtc_changing, что в свою очередь с задержкой в 1-2 такта pclk вызывает появление строба change_pulse. Этот строб служит разрешающим сигналом для записи регистров в домене pclk.

40.4 Сигналы блока

Таблица 40.1. Сигналы блока

Название	Направление	Описание
Тактовые сигналы и сигнал сброса		
rtc_clk	I	Тактовый сигнал RTC равный 32.768 кГц
rtc_reset_n	I	Асинхронный сигнал сброса внутренних регистров модуля *rtc_core* (текущие дату, время, счетчики интервального таймера, сигналы прерываний и счетчик количества прерываний интервального таймера). Активный 0.
pclk	I	Тактовый сигнал APB шины
presetn	I	Асинхронный сигнал сброса шины APB сбрасывает все регистры *pclk* домена, а также модуль write_fifo. Активный 0. Должен сниматься по положительному фронту *pclk*.
Сигналы шины APB		
penable	I	Сигнал разрешения
pwrite	I	Сигнал чтение/запись
pwdata[31:0]	I	Шина данных на запись
paddr[5:0]	I	Шина адреса
psel	I	Сигнал выбора периферийного устройства
prdata[31:0]	O	Шина данных на чтение
pready	O	Сигнал готовности периферийного устройства
Прочие сигналы		
pwrdn	I	Сигнал, сигнализирующий о переходе системы в режим *power down* (1 — *power down* активен, 0 — *power down* не активен)
alarm_intrpt	O	Сигнал прерывания по будильнику
it_intrpt	O	Сигнал периодического прерывания интервального таймера
wkup	O	Сигнал пробуждения системы (выход из режима *power down)*
test_mode	I	Сигнал управления DFT

40.5 Будильник

Функционал будильника реализован средствами двух регистров RRTC_TALRM и RRTC_DALRM. Установленные в этих регистрах значения полей будут сравниваться с соответствующими полями регистров даты и времени RRTC_TIME и RRTC_DATE. Разрешение сравнение по каждому типу поля выставляется отдельно. Например если разрешить сравнивать только шестнадцатые доли секунды, то будильник будет срабатывать каждую секунду. Если добавить в разрешение секунды, то каждую минуту и т. д. Если в регистре RRTC_CTRL установлен соответствующий разрешающий бит INT_ALRM_EN,

срабатывание будильника вызывает соответствующее прерывание: на выходе *alarm_intrpt* выставляется 1 и держится в течении одного такта *rtc_clk* (~30мкс). Независимо от разрешающего бита INT_ALRM_EN в момент срабатывания будильника в регистре статуса устанавливается бит INT_ALRM. Очистка статусного бита происходит при записи в этот бит единицы.

40.6 Интервальный таймер

Интервальный таймер, представляет собой два счетчика. Вспомогательный на 16 бит, и основной на 28 бит. Вспомогательный задает частоту работы основного. Максимальное значение вспомогательного счетчика определяется значением поля IT_MLT регистра RRTC_TCNT. Частоту работы основного счетчика рассчитывается следующим образом:

$f_{IT} = f_{RTC} / N$, где $N=2^{IT\_MLT}$,

При достижении основным счетчиком максимального значения, задаваемого в поле IT_VAL регистра RRTC_TCNT, значение этого счетчик сбрасывается и счет продолжается. Если в регистре RRTC_CTRL установлен соответствующий разрешающий бит INT_IT_EN, то в момент сброса основного счетчика происходит прерывание интервального таймера: на выходе *it_intrpt* выставляется 1 и держится в течение 1-го такта *rtc_clk* (~30мкс). Независимо от разрешающего бита INT_IT_EN в регистре статуса устанавливается соответствующий бит INT_IT. Очистка статусного бита происходит при записи в этот бит единицы.

Период работы интервального таймера рассчитывается по следующей формуле:

$T_{IT} = IT\_VAL / f_{IT}$;

Значение IT_VAL не может быть задаваться равным 0. Совместная установка IT_VAL = 1 и IT_MLT = 0 также не корректна. Некорректно установленные значения игнорируются.

Значения частоты работы интервального таймера и максимально отсчитываемые периоды (при IT_VAL=28'hFFF_FFFF) в зависимости от значения поля IT_MLT, представлены в следующей таблице:

Таблица 40.2.

Значение регистра IT_MLT	Частота работы основного счетчика интервального таймера, (Гц)	Максимальный возможный период таймера, (сек)
b0000	32768 Гц	~8192
b0001	16384 Гц	~16384
b0010	8192 Гц	~32768
...	...	...
b1101	4 Гц	~67108864
b1110	2 Гц	~134217728
b1111	1 Гц	~268435456 ≈ 8.5 лет

При чтении регистра RRTC_TCUR возвращается текущее значение основного счетчика IT_TCUR и количество прерываний интервального таймера IT_INT_CNT. Значения обоих счетчиков интервального таймера сбрасываются во время операции записи любого из полей регистра RRTC_TCNT. Максимальное значение количества прерываний, хранимое в поле IT_INT_CNT равно 15. При превышении этого числа, счетчик будет сбрасываться и начинать счет сначала. Количество прерываний также сбрасывается каждый раз при очистке бита INT_IT в регистре статуса.

40.7 RTC таймер и режим сна микросхемы

Архитектура блока предусматривает возможность индикации событий будильника или интервального таймера на вывод RTC_WAKEUP для вывода микросхемы из режима сна. Регистр RRTC_CTRL содержит биты ALRM_WKUP_EN и IT_WKUP_EN разрешающие эту возможность. Если один из этих бит установлен, то срабатывание будильника либо интервального таймера на следующем такте rtc_clk вызовет появление на выходе RTC_WAKUP высокого уровня сигнала. Сразу после вывода системы из режима сна (после подачи питания, снятия сигнала сброса и сигнала RTC_ISO) в регистре статуса произойдет установка битов ALARM_WKUP или IT_WKUP (в зависимости от того, чем был вызвано событие). Очистка статусных бит происходит при записи в них единицы. Снятие сигнала RTC_WAKEUP произойдет сразу после снятия сигнала RTC_ISO микросхемы.

40.8 Описание регистров

Таблица 40.3. Описание аббревиатур, используемых при описании регистров

Сокращение	Описание
–	Поле не доступно для записи, операция чтения возвращает 0.
RO	Поле доступно только для чтения, операция записи игнорируется.
WO	Поле доступно только для записи, операция чтения возвращает 0.
RW	Поле доступно как для записи, так и для чтения.
RW1C	Поле доступно как для записи, так и для чтения. Запись 1 очищает поле. Запись 0 игнорируется.

В следующей таблице кратко представлено описание всех регистров, реализованных в адресном пространстве блока ELRTC1:

Таблица 40.4. Общая карта регистров RTC

Адрес	Название	Описание
h00	RRTC_ID	Регистр идентификации
h04	RRTC_CTRL	Регистр управления.
h08	RRTC_TIME	Регистр, содержащий данные о времени.
h0C	RRTC_DATE	Регистр, содержащий данные о дате.
h10	RRTC_TALRM	Регистр, содержащий данные о времени будильника.
h14	RRTC_DALRM	Регистр, содержащий данные о дате будильника.
h18	RRTC_STAT	Регистр статуса прерываний.

h1C	RRTC_TCNT	Регистр настройки интервального таймера.
h20	RRTC_TCUR	Регистр текущего значения интервального таймера.
h24	RRTC_EXT_0	Дополнительный регистр расширения №0
h28	RRTC_EXT_1	Дополнительный регистр расширения №1
h2C	RRTC_EXT_2	Дополнительный регистр расширения №2
h30	RRTC_EXT_3	Дополнительный регистр расширения №3

С момента установки системного сброса *presetn* в 0 и на протяжении 2-х периодов сигнала *rtc_clk* (~60мкс) модуль *write_fifo* находится в состоянии сброса. В следствии этого запись во внутренние регистры модуля *rtc_core* невозможна. Регистр RRTC_STAT находится в частотном домене *pclk*, запись в него доступна непосредственно после поднятия *presetn* и происходит моментально. Изменение значений остальных регистров после совершения операции записи происходит с задержкой, обусловленной передачей данных через FIFO. При одной операции записи, значение регистра изменится через 2 полных периода сигнала *rtc_clk* (~60мкс), отсчитываемых от первого положительного фронта *rtc_clk* после момента записи. Если производится последовательно N операций записи, то время передачи увеличится на 1 такт *rtc_clk* (~30мкс) для каждого дополнительного регистра, т.е. займет N+1 полных периодов *rtc_clk*. Чтение регистров происходит моментально. Асинхронный сигнал сброса *rtc_reset_n* прежде чем поступить в rtc_core синхронизируется на частоте rtc_clk. Поэтому снятие сброса для модуля *rtc_core* произойдет только спустя 2 такта rtc_clk после подачи активного уровня сигнала на вход *rtc_reset_n*. На протяжении этого времени при чтении по шине APB будет возвращаться значение h45524F52 («EROR» в ASCII коде).

Проверка корректности ввода данных в регистры RRTC_TIME и RRTC_DATE никак не реализована. В случае записи некорректного значения в какое-либо поле этих регистров, счетчики работающие со значениями остальных полей будут работать без изменений, но при появлении ситуации, требующей инкрементирования некорректно введенного регистра, значение этого регистра сбросится на следующее корректное по смыслу. Например, при установке 21 часа и 78 минут, блок ELRTC1 досчитает текущие единицы минут до конца (т. е. до 9). Затем, в момент переключения некорректно введенного значения десятков минут, в соответствующих полях регистра RTC_TIME установится 22 часа и 00 минут. Последовательность прочитанных данных в этом случае будет выглядеть следующим образом:

… 21 час 78 минут 00 секунд...

…

… 21 час 78 минут 59 секунд...

… 21 час 79 минут 00 секунд...

…

… 21 час 79 минут 59 секунд...

… 22 час 00 минут 00 секунд...

В случае отключения тактового сигнала pclk достоверное значение внутренних регистров модуля rtc_core будет доступно либо после сброса presetn, либо после изменения значения одного из внутренних регистров (записано новое значение в какой либо регистр, переключились шестнадцатые доли секунды, секунды, минуты и пр., изменилось значение основного счетчика интервального таймера).

40.8.1 Описание регистра RRTC_ID

При чтении из этого регистра, возвращается значение идентификационного регистра.

Таблица 40.5. Поля регистра RRTC_ID

Номер бита	Поле	Тип	Описание	Сброс
31:8	NAME	R0	«RTC» в кодировке ASCII	24'h525443
7:4	–	R0	Резерв	4'h0
3:0	NVER	R0	Номер версии блока ELRTC1	4'h1

40.8.2 Описание регистра RRTC_CTRL

Регистр RRTC_CTRL управляет функционированием ядра RTC.

Таблица 40.6. Поля регистра RRTC_CTRL

Номер бита	Поле	Тип	Описание	Сброс
0	INT_ALRM_EN	RW	Если бит установлен, то разрешено появление прерывания на выходе *alarm_intrpt* по срабатыванию будильника.	1'h0
1	INT_IT_EN	RW	Если бит установлен, то разрешено появление прерывания на выходе *it_intrpt* по событию от интервального таймера.	1'h0
2	ALRM_WKUP_EN	RW	Если бит установлен, то разрешен вывод системы из режима power down по срабатыванию будильника.	1'h0
3	IT_WKUP_EN	RW	Если бит установлен, то разрешен вывод системы из режима power down по событию от интервального таймера.	1'h0
7:4	USED_CELLS	R0	Счетчик занятых ячеек FIFO. Счетчик уменьшается по мере считывания данных в домене rtc_clk. Если значение счетчика равно 0, то все данные из FIFO переписаны в соответствующие внутренние регистры rtc_core.	4'h0
31:4	–	RO	Резерв	28'h0

40.8.3 Описание регистра RRTC_TIME

Регистр RRTC_TIME содержит несколько BCD счетчиков, составляющих текущее время. Запись в этот регистр установит соответствующие значения в эти счетчики. При чтении

регистра возвращается текущее время. Предделитель шестнадцатых долей секунды сбрасывается во время записи поля SXTH.

Таблица 40.7. Поля регистра RRTC_TIME

Номер бита	Поле	Тип	Описание	Сброс
3:0	SXTH	RW	Шестнадцатые доли секунды. Счетчик изменяется от 0 до 15.	4'h0
7:4	SEC	RW	Секунды. Счетчик изменяется от 0 до 9.	4'h0
10:8	TSEC	RW	Десятки секунд. Счетчик изменяется от 0 до 5.	3'h0
14:11	MIN	RW	Минуты. Счетчик изменяется от 0 до 9.	4'h0
17:15	TMIN	RW	Десятки минут. Счетчик изменяется от 0 до 5.	3'h0
21:18	HOUR	RW	Часы. Счетчик изменяется от 0 до 9 (при THOUR = 2, HOUR = 0...3)	4'h0
23:22	THOUR	RW	Десятки часов. Счетчик изменяется от 0 до 2.	2'h0
26:24	DOW	RW	День недели. Счетчик изменяется от 1 до 7. Не синхронизируется с календарем.	3'h6
27	MSXTH	WO	Маска шестнадцатых долей секунды. Если бит равен 1, то при записи поле SXTH будет изменено.	1'h0
28	MSEC	WO	Маска секунд . Если бит равен 1, то во время записи поля SEC и TSEC будут изменены.	1'h0
29	MMIN	WO	Маска минут . Если бит равен 1, то во время записи поля MIN и TMIN будут изменены.	1'h0
30	MHOUR	WO	Маска часов . Если бит равен 1, то во время записи поля HOUR и THOUR будут изменены.	1'h0
31	MDOW	WO	Маска дней недели. Если бит равен 1, то во время записи поле DOW будет изменено.	1'h0

40.8.4 Описание регистра RRTC_DATE

Регистр RRTC_DATE содержит несколько BCD счетчиков, составляющих текущую дату. Запись в этот регистр установит соответствующие значения в эти счетчики. При чтении регистра возвращается текущая дата.

Таблица 40.8. Поля регистра RRTC_DATE

Номер бита	Поле	Тип	Описание	Сброс
3:0	DAY	RW	Дни. Счетчик изменяется от 1 до 9 при TDAY=0. Если TDAY = 1 - от 0 до 9. Если TDAY = 2 и текущий месяц февраль и год не високосный, DAY изменяется от 0 до 8, иначе от 0 до 9 . При TDAY = 3, DAY принимает значения 0 или 1 в зависимости от количества дней в текущем месяце (30 или 31).	4'h1
5:4	TDAY	RW	Десятки дней. Счетчик изменяется от 0 до 3. Исключение составляет Февраль, в этом случае максимальное значение TDAY = 2.	2'h0
9:6	MON	RW	Месяцы. Счетчик изменяется от 1 до 9 при TMON = 0, и от 0 до 2, если TMON = 1	4'h1
10	TMON	RW	Десятки месяцев. Счетчик изменяется от 0 до 1.	1'h0
14:11	YEAR	RW	Годы. Счетчик изменяется от 0 до 9.	4'h0
18:15	TYEAR	RW	Десятки лет. Счетчик изменяется от 0 до 9.	4'h0
22:19	CEN	RW	Сотни лет (века). Счетчик изменяется от 0 до 9.	4'h0
26:23	TCEN	RW	Тысячи лет (десятки веков). Счетчик изменяется от 0 до 9.	4'h2

Номер бита	Поле	Тип	Описание	Сброс
27	MDAY	WO	Маска дней. Если бит равен 1, то во время записи поля DAY и TDAY будут изменены.	1'h0
28	MMON	WO	Маска месяцев. Если бит равен 1, то во время записи поля MON и TMON будут изменены.	1'h0
29	MYEAR	WO	Маска лет. Если бит равен 1, то во время записи поля YEAR и TYEAR будут изменены.	1'h0
30	MCEN	WO	Маска веков. Если бит равен 1, то во время записи поля CEN и TCEN будут изменены.	1'h0
31	–	RO	Резерв	1'h0

40.8.5 Описание регистра RRTC_TALRM

Регистр RRTC_TALRM содержит поля, которые сравниваются с соответствующими полями регистра RRTC_TIME. При их совпадении если установлен разрешающий бит в регистре RRTC_CTRL возникает прерывание будильника.

Таблица 40.9. Поля регистра RRTC_TALRM

Номер бита	Поле	Тип	Описание	Сброс
3:0	SXTH	RW	Шестнадцатые доли секунды. Счетчик изменяется от 0 до 15.	4'h0
7:4	SEC	RW	Секунды. Принимает значения от 0 до 9.	4'h0
10:8	TSEC	RW	Десятки секунд. Принимает значения от 0 до 5.	3'h0
14:11	MIN	RW	Минуты. Принимает значения от 0 до 9.	4'h0
17:15	TMIN	RW	Десятки минут. Принимает значения от 0 до 5.	3'h0
21:18	HOUR	RW	Часы. Принимает значения от 0 до 9 (при THOUR = 2, HOUR = 0...3)	4'h0
23:22	THOUR	RW	Десятки часов. Принимает значения от 0 до 2.	2'h0
26:24	DOW	RW	День недели. Принимает значения от 1 до 7.	3'h0
27	CSXTH	RW	Когда бит установлен, шестнадцатые доли секунды из RRTC_TIME сравнивается с шестнадцатыми долями секунд будильника из RRTC_TALRM.	1'h0
28	CSEC	RW	Когда бит установлен, секунды сравнивается с секундами будильника.	1'h0
29	CMIN	RW	Когда бит установлен, минуты сравнивается с минутами будильника.	1'h0
30	CHOUR	RW	Когда бит установлен, часы сравнивается с часами будильника.	1'h0
31	CDOW	RW	Когда бит установлен, день недели сравнивается с днем недели будильника.	1'h0

40.8.6 Описание регистра RRTC_DALRM

Регистр RRTC_DALRM содержит поля, которые сравниваются с соответствующими полями регистра RRTC_DATE. При их совпадении если установлен разрешающий бит в регистре RRTC_CTRL возникает прерывание будильника.

Таблица 40.10. Поля регистра RRTC_DALRM

Номер бита	Поле	Тип	Описание	Сброс
3:0	DAY	RW	Дни. Аналогично таблице 5.	4'h0
5:4	TDAY	RW	Десятки дней. Аналогично таблице 5.	2'h0
9:6	MON	RW	Месяцы. Аналогично таблице 5.	4'h0

Номер бита	Поле	Тип	Описание	Сброс
10	TMON	RW	Десятки месяцев. Аналогично таблице 5.	1'h0
14:11	YEAR	RW	Годы. Принимает значения от 0 до 9.	4'h0
18:15	TYEAR	RW	Десятки лет. Принимает значения от 0 до 9.	4'h0
22:19	CEN	RW	Сотни лет (века). Принимает значения от 0 до 9.	4'h0
26:23	TCEN	RW	Тысячи лет (десятки веков). Принимает значения от 0 до 9.	4'h0
27	CDAY	RW	Когда бит установлен, дни из RRTC_DATE сравниваются с днями будильника из RRTC DALRM.	1'h0
28	CMON	RW	… сравниваются месяцы	1'h0
29	CYEAR	RW	… сравниваются годы (годы + десятилетия)	1'h0
30	CCEN	RW	… сравниваются века (столетия + тысячелетия)	1'h0
31	–	RO	Резерв	1'h0

40.8.7 Описание регистра RRTC_STAT

При чтении из этого регистра, возвращается набор статусов соответствующих прерываний. Очистка статусов прерываний производится записью единицы в соответствующий бит данного регистра. Запись нуля игнорируется.

Таблица 40.11. Поля регистра RRTC_STAT

Номер бита	Поле	Тип	Описание	Сброс
0	INT_ALRM	RW1C	Если бит установлен, то произошло прерывание от будильника.	1'h0
1	INT_IT	RW1C	Если бит установлен, то произошло прерывание от интервального таймера.	1'h0
2	ALRM_WKUP	RW1C	Если бит установлен, то произошло срабатывание будильника во время активного режима *power down*.	1'h0
3	IT_WKUP	RW1C	Если бит установлен, то произошло срабатывание интервального таймера во время активного режима *power down*.	1'h0
4	RTC_RESET	RW1C	Бит устанавливается в 1 при появлении активного сигнала сброса rtc_reset_n.	1'h0
5	APB_RESET	RW1C	Бит устанавливается в 1 при появлении активного сигнала сброса *presetn*.	1'h1
31:6	–	RO	Резерв	28'h0

40.8.8 Описание регистра RRTC_TCNT

Регистр RRTC_TCNT управляет работой интервального таймера.

Таблица 40.12. Поля регистра RRTC_TCNT

Номер бита	Поле	Тип	Описание	Сброс
27:0	IT_VAL	RW	Максимальное значение счетчика тактов, при достижении которого будет происходить прерывание. Установка в этом поле 0 игнорируется. Также игнорируется значение 1 при IT_MLT = 0.	28'h0
31:28	IT_MLT	RW	Значение, определяющее величину делителя частоты для интервального таймера.	4'h0

40.8.9 Описание регистра RRTC_TCUR

При чтении из этого регистра, возвращается текущее значение основного счетчика интервального таймера и количество произошедших прерываний.

Таблица 40.13. Поля регистра RRTC_TCUR

Номер бита	Поле	Тип	Описание	Сброс
27:0	IT_TCUR	RO	Текущее значение основного счетчика интервального таймера. Значения основного и вспомогательного счетчиков сбрасываются во время записи любого из полей регистра RRTC_TCNT.	28'h0
31:28	IT_INT_CNT	RO	Количество прерываний интервального таймера. Сбрасывается во время очищения бита INT_IT регистра статуса и по активному сигналу сброса *presetn*.	4'h0

40.8.10 Описание регистра RRTC_EXT_0... RRTC_EXT_3

Данные регистры являются RW регистрам общего назначения.

Таблица 40.14. Поля регистра RRTC_EXT_0... RRTC_EXT_3

Номер бита	Поле	Тип	Описание	Сброс
31:0	EXTENSION	RW		32’bX

41. БЛОК ОБМЕНА СООБЩЕНИЯМИ (MAILBOX)

41.1 Общая информация

41.1.1 Назначение

Mailbox позволяет программно организовать канал передачи данных между CPU и DSP. Структура модуля Mailbox представлена на Рисунок 41.1.

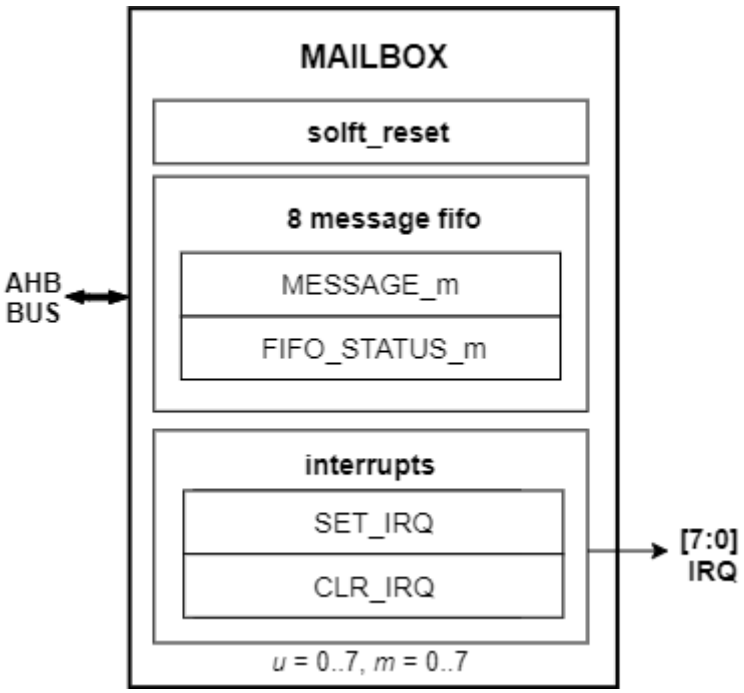


Рисунок 41.1. Структура модуля Mailbox

41.1.2 Характеристики

Блок обмена сообщениями имеет следующие характеристики:

- имеет 8 буферов FIFO_m (m=0…7), глубина – 8 слов, разрядность – 32;
- поддерживает режимы работы по прерыванию и по опросу;
- формирует 8 прерываний IRQ_u (u=0…7);
- установка и сброс прерываний осуществляется через отдельные регистры.

41.2 Регистры

В Таблица 41.1 приведен перечень программно-доступных регистров блока Mailbox.

Таблица 41.1. Перечень программно-доступных регистров Mailbox

Условное обозначение регистра	Название регистра	Тип доступа	Исходное состояние	Смещение
SOFT_RESET	Регистр программного сброса	W	0x0	0x8000
MB_VERSION	Регистр версии блока.	R	0x1	0x8004
MESSAGE_*m*	Регистр обмена	W/R	0x0	*m*=0: 0x0000 *m*=1: 0x1000 *m*=2: 0x2000 *m*=3: 0x3000 *m*=4: 0x4000 *m*=5: 0x5000 *m*=6: 0x6000 *m*=7: 0x7000
FIFO_STATUS_*m*	Регистр статуса FIFO	R	0x2	*m*=0: 0x0004 *m*=1: 0x1004 *m*=2: 0x2004 *m*=3: 0x3004 *m*=4: 0x4004 *m*=5: 0x5004 *m*=6: 0x6004 *m*=7: 0x7004
SET_IRQ_*u*	Регистр установки прерывания IRQ_u.	W/R	0x0	*u*=0 (mb0_rrq_risc0): 0x0014 *u*=1 (mb0_rrq_risc1): 0x1014 *u*=2 (mb0_rrq_dsp0): 0x2014 *u*=3 (mb0_rrq_dsp1): 0x3014 *u*=4 (mb0_rrq_cpu0): 0x4014 *u*=5 (mb0_rrq_cpu1): 0x5014 *u*=6 (mb0_rrq_cpu2): 0x6014 *u*=7 (mb0_rrq_cpu3): 0x7014
CLR_IRQ_*u*	Регистр сброса прерывания IRQ_u.	W/R	0x0	*u*=0 (mb0_rrq_risc0): 0x001C *u*=1 (mb0_rrq_risc1): 0x101C *u*=2 (mb0_rrq_dsp0): 0x201C *u*=3 (mb0_rrq_dsp1): 0x301C *u*=4 (mb0_rrq_cpu0): 0x401C *u*=5 (mb0_rrq_cpu1): 0x501C *u*=6 (mb0_rrq_cpu2): 0x601C *u*=7 (mb0_rrq_cpu3): 0x701C

41.2.1 Регистр SOFT_RESET

Регистр программного сброса.

Формат регистра SOFT_RESET приведен в Таблица 41.2.

Таблица 41.2. Формат регистра SOFT_RESET

Номер бита	Условное обозначение	Назначение
0	SOFT_RESET	Программный сброс обнуляет содержимое буферов, регистров статуса FIFO и все прерывания.
1:31	-	Резерв

41.2.2 Регистр MB_VERSION

Регистр версии блока.

Формат регистра MB_VERSION приведен в Таблица 41.3.

Таблица 41.3. Формат регистра MB_VERSION

Номер бита	Условное обозначение	Назначение
31:0	MB_VERSION	Регистр содержит номер версии блока MAILBOX.

41.2.3 Регистр MESSAGE_*m*

Регистр чтения/записи в FIFO_*m*.

Формат регистра MESSAGE_*m* приведен вТаблица 41.4.

Таблица 41.4. Формат регистра MESSAGE_*m*

Номер бита	Условное обозначение	Назначение
0:31	MESSAGE	Регистр, через который происходит чтение и запись данных в Mailbox.

41.2.4 Регистр FIFO_STATUS_m

Регистр статуса FIFO_*m*.

Формат регистра FIFO_STATUS_*m* приведен в Таблица 41.5.

Таблица 41.5. Формат регистра FIFO_STATUS_*m*

Номер бита	Условное обозначение	Назначение
0	FULL	Регистр, указывающий на то, что FIFO полон.
1	EMPTY	Регистр, указывающий на то, что FIFO пуст.
4:7	STATUS	Регистр, указывающий на число слов данных в FIFO.
8:31	-	Резерв

41.2.5 Регистр SET_IRQ_u

Регистр включения прерывания IRQ_*u*. Прерывание IRQ_*u* формируется сложением всех бит этого регистра по ИЛИ. Каждому из 8-и бит этого регистра соответствует буфер FIFO. Значение регистра указывает, из какого буфера ожидается чтение данных.

Формат регистра SET_IRQ_*u* приведен в Таблица 41.6.

Таблица 41.6. Формат регистра SET_IRQ_*u*

Номер бита	Условное обозначение	Назначение
0:7	SET_IRQ_*u*	Для каждого бита:

Номер бита	Условное обозначение	Назначение
		Чтение 0: прерывание не установлено. Чтение 1: прерывание установлено. Запись 0: игнорируется Запись 1: установить прерывание.
8:31	-	Резерв

41.2.6 Регистр CLR_IRQ_*u*

Регистр сброса прерывания IRQ_*u*. Запись в любой из 8-и бит этого регистра, обнуляет значение, установленное в соответствующем бите регистра IRQ_*u*.

Формат регистра CLR_IRQ_*u* приведен в Таблица 41.7.

Таблица 41.7. Формат регистра CLR_IRQ_*u*

Номер бита	Условное обозначение	Назначение
0:7	CLR_IRQ_*u*	Для каждого бита: Чтение 0: прерывание не установлено. Чтение 1: прерывание установлено. Запись 0: игнорируется Запись 1: убрать прерывание.
8:31	-	Резерв

41.3 Методы организации канала межпроцессорного обмена

41.3.1 По прерыванию IRQ_*u*

Если некоторый блок IP0 имеет данные на передачу другому блоку IP1, то IP0 необходимо записать данные в пустой FIFO_*m* (при этом FIFO_STATUS_*m*[1]==1) через регистр MESSAGE_*m* и записать 1 в бит SET_IRQ_*u*[*m*], где номер прерывания *u* соответсвюет IP1, что приведет к запросу прерывания IRQ_*u* на чтение IP1 из FIFO_*m*.

По результатам обработки прерывания, IP1 необходимо прочитать регистр SET_IRQ_*u*, для определения номера FIFO_*m*, содержащего данные, считать данные из FIFO_*m* через регистр MESSAGE_*m*, а после сбросить прерывание, записав 1 в регистр CLR_IRQ_*u*[*m*].

41.3.2 По опросу

Если IP0 необходимо периодически передавать данные IP1 в выбранный FIFO_*m*, то IP0 необходимо записывать данные через регистр MESSAGE_*m*, проверяя содержимое регистра FIFO_STATUS_*m* на наличие свободного места в FIFO_*m*.

Соответственно, IP1 должно периодически опрашивать регистр FIFOSTATUS_*m* на наличие в нем данных, и читать их через регистр MESSAGE_*m*.

42. БЛОК АТОМАРНЫХ ОПЕРАЦИЙ (SPINLOCK)

Блок Spinlock представляет собой набор из 512-и низкоуровневых взаимоисключающих примитивов синхронизации, каждый из которых — однобитный регистр. Для их использования необходимо выполнять цикл ожидания получения блокировки.

42.1 Регистры Spinlock

Таблица 42.1.

Условное обозначение регистра	Название регистра	Тип доступа	Исходное состояние	Смещение
LOCK_*n* (*n* = 0..511)	Регистр блокировки	W/R	0x0	0x0 + 0x4**n*
CHECK_*n* (*n*=0..511)	Регистр проверки блокировки	R	0x0	0x800 + 0x4**n*
SOFT_RESET	Регистр программного сброса.	W	0x0	0x1000

42.2 Алгоритм работы

Каждый однобитный регистр LOCK_n имеет два состояния:

- NOT TAKEN (0)
- TAKEN (1)

При этом переход NOT TAKEN -> TAKEN происходит при чтении бита, находящегося в состоянии NOT TAKEN. А переход TAKEN -> NOT TAKEN происходит при записи 0 в бит TAKEN.

Алгоритм работы модуля представлен на Рисунок 42.1.

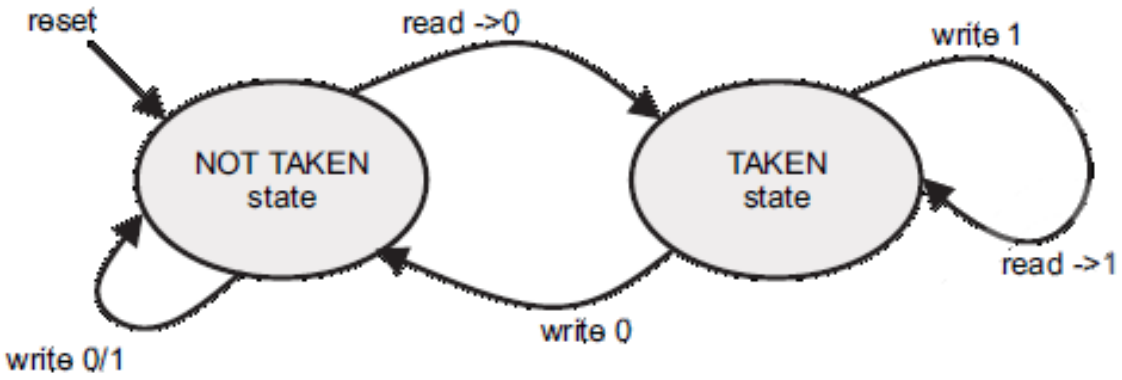


Рисунок 42.1. Алгоритм переключения LOCK регистров

Любой регистр CHECK_*n* позволяет проверить состояние блокировки соответствующего LOCK_*n* регистра без изменения его состояния NOT TAKEN -> TAKEN.

Запись 1 в регистр SOFT_RESET сбрасывает все регистры LOCK_*n* в состояние NOT_TAKEN(0).

43. УНИВЕРСАЛЬНЫЙ БЛОК ФОРМИРОВАНИЯ ТАКТОВОЙ ЧАСТОТЫ (UCG)

43.1 Общая информация

Блок UCG предназначен для формирования «веера» частот необходимых конкретной подсистеме. Имеет следующие особенности:

- выбор входной референсной частоты (одной из четырех)
- 16 независимых каналов формирования синхросигналов
- диапазон деления от 1 до 2^{20}-1.
- поддержка Low Power интерфейса типа Q-Channel.
- Bypass mode. Отключение делителя и трансляция входной частоты «clk_xti» на выход clk_o любого из каналов.
- Выравнивание фазы синхросигналов различных каналов

При работе с блоком необходимо помнить о том, что в микросхеме не предусмотрена защита от выключения частоты коммутатора подсистемы (как правило, принадлежит каналу 0 UCG). Выключение частоты коммутатора любой подсистемы приведет к ее зависанию.

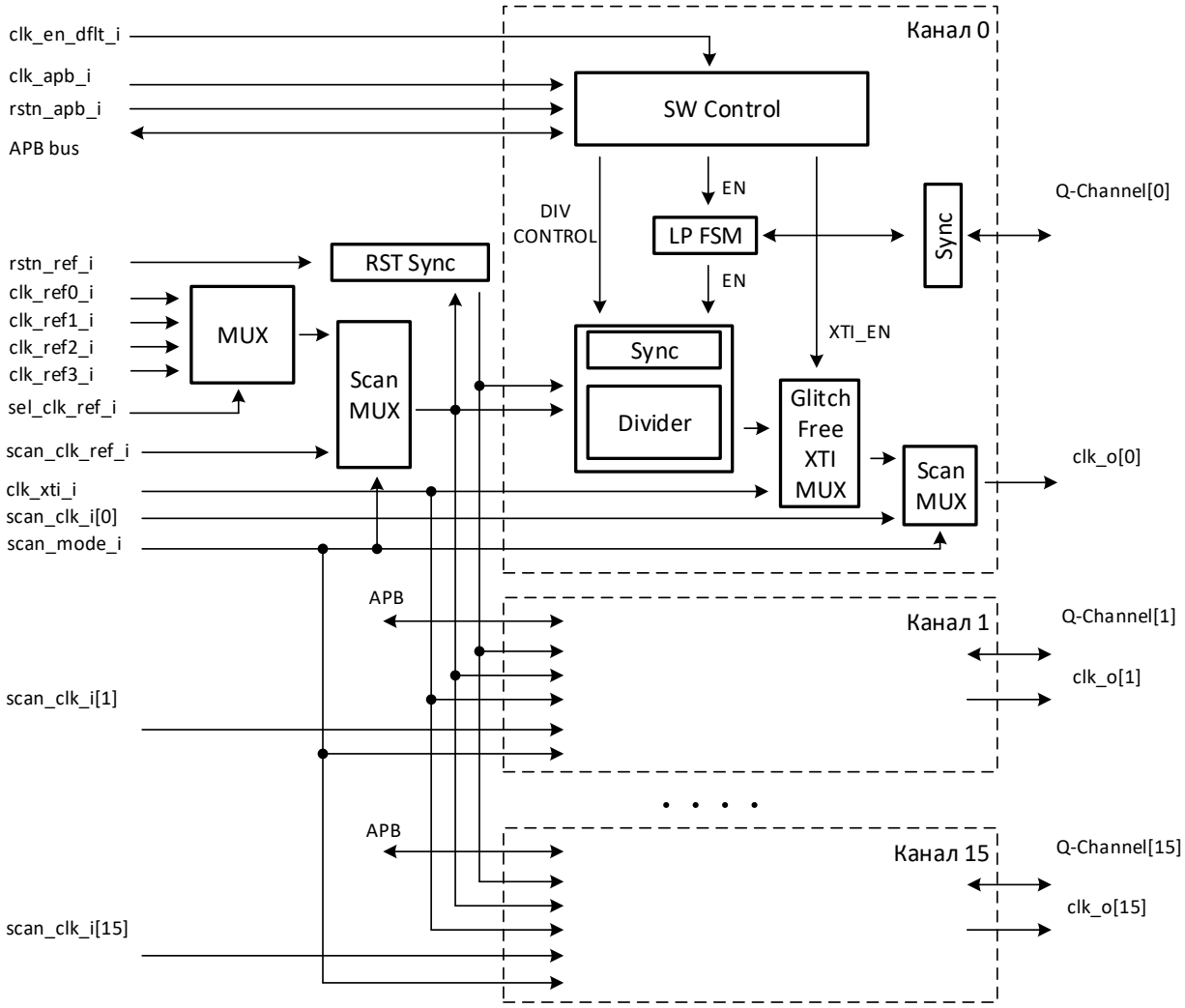


Рисунок 43.1. Структурная схема

43.1.1 Домены синхронизации

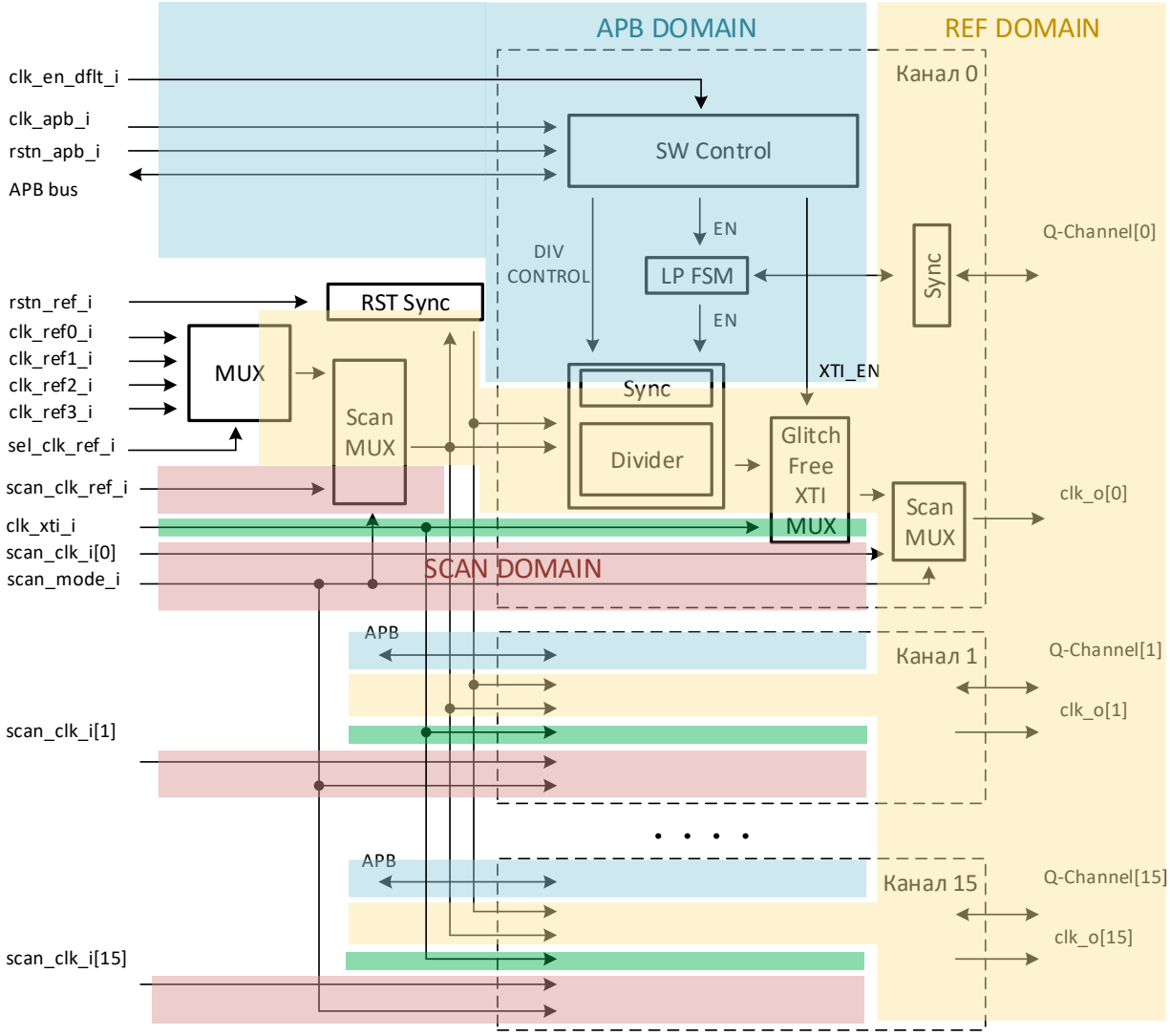


Рисунок 43.2. Домены синхронизации

43.1.2 Описание портов

Таблица 43.1. Описание портов

Название	Направление	Описание	Подключение
clk_en_dflt_i[15:0]	input	Tie off signal. Определяет значение по сбросу полей “LPI_EN” и “CLK_EN” регистра “UCG_CTR*_REG” clk_en_dflt_i[0] – регистра “UCG_CTR0_REG” clk_en_dflt_i[1] – регистра “UCG_CTR1_REG” и т.д.	
clk_xti_i	input	Частота XTI. Always on.	
clk_apb_i	input	Частота шины APB	
rstn_apb_i	input	Сброс APB. Выставляется асинхронно, снимается синхронно частоте APB. Активный уровень «0»	
paddr_i[31:0]	input	APB интерфейс	
psel_i	input		

Название	Направление	Описание	Подключение
penable_i	input		
pwrite_i	input		
pwdata_i[31:0]	input		
prdata_o[31:0]	output		
clk_ref0_i	input	Референсная частота 0	
clk_ref1_i	input	Референсная частота 1	
clk_ref2_i	input	Референсная частота 2	
clk_ref3_i	input	Референсная частота 3	
rst_ref_n_i	input	Сигнал сброса домена референсной частоты. Полностью асинхронный сигнал. Активный уровень «0»	
sel_clk_ref_i[1:0]	input	Выбор рефернсной частоты • 2'b00 – clk_ref0_i • 2'b01 – clk_ref1_i • 2'b10 – clk_ref2_i • 2'b11 – clk_ref3_i	
qreqn_o[N-1:0]	output	Q-Channel интерфейс N – количество каналов В каждом канале свой интерфейс	
qacceptn_i[N-1:0]	input		
qdeny_i[N-1:0]	input		
qactive_i[N-1:0]	input		
clk_o[N-1:0]	output	Выходная частота N – количество каналов	
scan_mode_i	input	Признак перехода в тестовый режим • 0 – функциональный режим • 1 – тестовый режим	
scan_clk_ref_i	input	Вход тестовой частоты для мультиплексии референсного синхросигнала	
scan_clk_i[N-1:0]	input	Входы тестовых частот N – количество каналов	

43.2 Регистры

Перечень регистров приведен в Таблица 43.2.

Таблица 43.2. Перечень программно-доступных регистров

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
UCG_CTR00_REG	Регистр управления каналом 0	0x40000001, если clk_en_dflt_i[0] = 0 0x40000302, если clk_en_dflt_i[0] = 1	W/R	0x00
UCG_CTR01_REG	Регистр управления каналом 1	0x40000001, если clk_en_dflt_i[1] = 0 0x40000302, если clk_en_dflt_i[1] = 1	W/R	0x04
UCG_CTR02_REG	Регистр управления каналом 2	0x40000001, если clk_en_dflt_i[2] = 0 0x40000302, если clk_en_dflt_i[2] = 1	W/R	0x08
UCG_CTR03_REG	Регистр управления каналом 3	0x40000001, если clk_en_dflt_i[3] = 0 0x40000302, если clk_en_dflt_i[3] = 1	W/R	0x0C
UCG_CTR04_REG	Регистр управления каналом 4	0x40000001, если clk_en_dflt_i[4] = 0 0x40000302, если clk_en_dflt_i[4] = 1	W/R	0x10
UCG_CTR05_REG	Регистр управления каналом 5	0x40000001, если clk_en_dflt_i[5] = 0 0x40000302, если clk_en_dflt_i[5] = 1	W/R	0x14
UCG_CTR06_REG	Регистр управления каналом 6	0x40000001, если clk_en_dflt_i[6] = 0 0x40000302, если clk_en_dflt_i[6] = 1	W/R	0x18

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
UCG_CTR07_REG	Регистр управления каналом 7	0x40000001, если clk_en_dflt_i[7] = 0 0x40000302, если clk_en_dflt_i[7] = 1	W/R	0x1C
UCG_CTR08_REG	Регистр управления каналом 8	0x40000001, если clk_en_dflt_i[8] = 0 0x40000302, если clk_en_dflt_i[8] = 1	W/R	0x20
UCG_CTR09_REG	Регистр управления каналом 9	0x40000001, если clk_en_dflt_i[9] = 0 0x40000302, если clk_en_dflt_i[9] = 1	W/R	0x24
UCG_CTR10_REG	Регистр управления каналом 10	0x40000001, если clk_en_dflt_i[10] = 0 0x40000302, если clk_en_dflt_i[10] = 1	W/R	0x28
UCG_CTR11_REG	Регистр управления каналом 11	0x40000001, если clk_en_dflt_i[11] = 0 0x40000302, если clk_en_dflt_i[11] = 1	W/R	0x2C
UCG_CTR12_REG	Регистр управления каналом 12	0x40000001, если clk_en_dflt_i[12] = 0 0x40000302, если clk_en_dflt_i[12] = 1	W/R	0x30
UCG_CTR13_REG	Регистр управления каналом 13	0x40000001, если clk_en_dflt_i[13] = 0 0x40000302, если clk_en_dflt_i[13] = 1	W/R	0x34
UCG_CTR14_REG	Регистр управления каналом 14	0x40000001, если clk_en_dflt_i[14] = 0 0x40000302, если clk_en_dflt_i[14] = 1	W/R	0x38
UCG_CTR15_REG	Регистр управления каналом 15	0x40000001, если clk_en_dflt_i[15] = 0 0x40000302, если clk_en_dflt_i[15] = 1	W/R	0x3C
UCG_BP_CTR_REG	Регистр управления переходом в режим BYPASS	0x0	W/R	0x40
UCG_SYNC_CLK_REG	Регистр синхронизации каналов	0x0	WO	0x44

43.2.1 Регистр UCG_CTR*_REG

Регистр UCG_CTR*_REG.

Формат регистра UCG_CTR*_REG приведен в Таблица 43.3.

Таблица 43.3. Формат регистра UCG_CTR*_REG

Номер бита	Условное обозначение	Назначение
0	LPI_EN	Бит включения автоматического режима работы LowPower интерфейса «Q-Channel» канала *: 0 – автоматический режим выключен 1 – автоматический режим включен **Важно:** Для UCG в подсистеме SDR режим LPI не поддерживается. Для всех регистров UCG_CTR*_REG бит LPI_EN=0. Тип доступа: RW Значение по сбросу: ~clk_en_dflt_i[*]
1	CLK_EN	Бит управления включением частоты канала *: 0 – частота выключена 1 – частота включена Тип доступа: RW Значение по сбросу: clk_en_dflt_i[*] Если блок работает в автоматическом режиме (LPI_EN = 1), то значение поля CLK_EN может меняться в процессе работы если: 1. Поступил SW запрос на отключение частоты, но подключенное устройство его отклонило. CLK_EN

Номер бита	Условное обозначение	Назначение
		изменит значение с 0 на 1 установится 1 в CLK_EN_STATUS[0] 2. Поступил запрос на отключение частоты со стороны подключенного устройства и QACTIVE_CTK_EN = 1. CLK_EN изменит значение с 1 на 0 установится 1 в CLK_EN_STATUS[1] 3. Поступил запрос на включение частоты со стороны подключенного устройства и QACTIVE_CTK_EN = 1. CLK_EN изменит значение с 0 на 1 установится 1 в CLK_EN_STATUS[2]
4:2	CLK_EN_STS	Отображение события, по которому был переключен бит CLK_EN CLK_EN_STATUS[0] = 1 - отключение частоты отклонено из-за занятости устройства (QDENY = 1) CLK_EN_STATUS[1] = 1 - выполнено отключение частоты по сингалу QACTIVE CLK_EN_STATUS[2] = 1 - выполнено включение частоты по сингалу QACTIVE Сброс статусных битов осуществляется записью "1" в соответствующий бит Тип доступа: RW Значение по сбросу: 3’b0
5	-	резерв
6	QACTIVE_CTL_EN	Бит разрешения реакции на сигнал QACTIVE (разрешение на изменения состояния со стороны подключенного устройства) при работе в LPI (LPI_EN = 1) режиме: 0 – реакция запрещена 1 – реакция разрешена Бит ни на что не влияет, если LPI режим выключен (LPI_EN = 0) Тип доступа: RW Значение по сбросу: 1’b0
9:7	Q_FSM_STATE	Биты отображения состояния LowPower Q_FSM : 110 - Q_RUN 010 - Q_REQUEST 000 - Q_STOPPED 100 - Q_EXIT 011 - Q_DENIED 111 - Q_CONTINUE 001 - Q_CLK_EN Тип доступа: RO Значение по сбросу: - 3’b000 - если clk_en_dflt_i[*] = 0 - 3’b110 - если clk_en_dflt_i[*] = 1
29:10	DIV_COEFF	Коэффициент деления входной частоты 0 – деление на 1 1 – деление на 1 2 – деление на 2 3 – деление на 3 … 2^{20}-1 – деление на 2^{20}-1 Тип доступа: RW Значение по сбросу: 20’b0
30	DIV_LOCK	Признак готовности делителя 0 - идет переключение на новый коэффициент деления 1 - переключение произведено Тип доступа: RO Значение по сбросу: 1’b1
31	-	резерв

43.2.2 Регистр UCG_BP_CTR_REG

Регистр UCG_BP_CTR_REG.

Формат регистра UCG_BP_CTR_REG приведен в Таблица 43.3.

Таблица 43.4. Формат регистра UCG_BP_CTR_REG

Номер бита	Условное обозначение	Назначение
0	XTI_EN_CH0	Включение Bypass mode канала 0. 0 – mission mode 1 – bypass mode Тип доступа: RW
1	XTI_EN_CH1	Включение Bypass mode канала 1. 0 – mission mode 1 – bypass mode Тип доступа: RW
2	XTI_EN_CH2	Включение Bypass mode канала 2. 0 – mission mode 1 – bypass mode Тип доступа: RW
3	XTI_EN_CH3	Включение Bypass mode канала 3. 0 – mission mode 1 – bypass mode Тип доступа: RW
4	XTI_EN_CH4	Включение Bypass mode канала 4. 0 – mission mode 1 – bypass mode Тип доступа: RW
5	XTI_EN_CH5	Включение Bypass mode канала 5. 0 – mission mode 1 – bypass mode Тип доступа: RW
6	XTI_EN_CH6	Включение Bypass mode канала 6. 0 – mission mode 1 – bypass mode Тип доступа: RW
7	XTI_EN_CH7	Включение Bypass mode канала 7. 0 – mission mode 1 – bypass mode Тип доступа: RW
8	XTI_EN_CH8	Включение Bypass mode канала 8. 0 – mission mode 1 – bypass mode Тип доступа: RW
9	XTI_EN_CH9	Включение Bypass mode канала 9. 0 – mission mode 1 – bypass mode Тип доступа: RW
10	XTI_EN_CH10	Включение Bypass mode канала 10. 0 – mission mode 1 – bypass mode Тип доступа: RW
11	XTI_EN_CH11	Включение Bypass mode канала 11. 0 – mission mode 1 – bypass mode Тип доступа: RW

Номер бита	Условное обозначение	Назначение
12	XTI_EN_CH12	Включение Bypass mode канала 12. 0 – mission mode 1 – bypass mode Тип доступа: RW
13	XTI_EN_CH13	Включение Bypass mode канала 13. 0 – mission mode 1 – bypass mode Тип доступа: RW
14	XTI_EN_CH14	Включение Bypass mode канала 14. 0 – mission mode 1 – bypass mode Тип доступа: RW
15	XTI_EN_CH15	Включение Bypass mode канала 15. 0 – mission mode 1 – bypass mode Тип доступа: RW
31:16	-	резерв

43.2.3 Регистр UCG_SYNC_CLK_REG

Регистр UCG_SYNC_CLK_REG.

Формат регистра UCG_SYNC_CLK_REG приведен в Таблица 43.3.

Таблица 43.5. Формат регистра UCG_SYNC_CLK_REG

Номер бита	Условное обозначение	Назначение
0	SYNC_CLK_CH0	Запрос на синхронизацию канала 0. Активный уровень «1»
1	SYNC_CLK_CH1	Запрос на синхронизацию канала 1. Активный уровень «1»
2	SYNC_CLK_CH2	Запрос на синхронизацию канала 2. Активный уровень «1»
3	SYNC_CLK_CH3	Запрос на синхронизацию канала 3. Активный уровень «1»
4	SYNC_CLK_CH4	Запрос на синхронизацию канала 4. Активный уровень «1»
5	SYNC_CLK_CH5	Запрос на синхронизацию канала 5. Активный уровень «1»
6	SYNC_CLK_CH6	Запрос на синхронизацию канала 6. Активный уровень «1»
7	SYNC_CLK_CH7	Запрос на синхронизацию канала 7. Активный уровень «1»
8	SYNC_CLK_CH8	Запрос на синхронизацию канала 8. Активный уровень «1»
9	SYNC_CLK_CH9	Запрос на синхронизацию канала 9. Активный уровень «1»
10	SYNC_CLK_CH10	Запрос на синхронизацию канала 10. Активный уровень «1»
11	SYNC_CLK_CH11	Запрос на синхронизацию канала 11. Активный уровень «1»
12	SYNC_CLK_CH12	Запрос на синхронизацию канала 12. Активный уровень «1»
13	SYNC_CLK_CH13	Запрос на синхронизацию канала 13. Активный уровень «1»
14	SYNC_CLK_CH14	Запрос на синхронизацию канала 14. Активный уровень «1»
15	SYNC_CLK_CH15	Запрос на синхронизацию канала 15. Активный уровень «1»
31:16	-	резерв

43.3 Функционирование

43.3.1 Последовательность выхода из состояния сброса

1. Установка стабильной частоты clk_apb_i и clk_ref*_i
2. Снятие сброса rst_apb_i

3. Выбор референсной частоты
4. Снятие сигнала сброса rst_ref_i
5. Установка коэффициентов деления

43.3.2 Выбор референсной частоты

В блок UCG можно завести до 4-х референсных частот «clk_ref*_i». Выбор референсной частоты, от которой будет работать весь блок, осуществляется при помощи сигнала «sel_clk_ref_i». В блоке не предусмотрен плавный переход с одной референсной частоты на другую, поэтому изменение значения сигнала «sel_clk_ref_i» возможно только при соблюдении одного из следующих условий:

1. Блок находится под активным сигнала сброса «rst_ref_n_i».
2. Все каналы выключены или переведены в bypass mode.

Для большинства подсистем не требуется смена референсной частоты, в этом случае на «sel_clk_ref_i» должна быть заведена константа.

43.3.3 Установка коэффициентов деления

После выбора референсной частоты и снятия сигнала сброса «rst_ref_n_i» можно приступать к программированию делителей.

В дефолтном состоянии блок находится в mission mode, деление входного синхросигнала не производится, состояние выходного «ClockGate» зависит от значения на входе “clk_en_dflt_i[*]”.

Для включения нужного канала (если “clk_en_dflt_i[*]” = 0) необходимо провести следующие действия:

1. Записать нужный коэффициент деления UCG_CTR*_REG[DIV_COEFF], если требуется его изменить
2. Дождаться перехода на новый коэффициент деления (в поле DIV_LOCK регистра UCG_CTR*_REG должна появиться 1).
3. Разблокировать выходной «ClockGate» (включить частоту). Разблокировка возможна как в автоматическом (с поддержкой Q-Channel интерфейса), так и в ручном режиме.

43.3.4 Изменение коэффициента деления

При изменении коэффициента деления включенного делителя возможно формирование на выходе коротких импульсов длительностью в половину периода референсной частоты. Если это критично для данного частотного домена, то переход на новую частоту следует осуществлять по следующему алгоритму:

1. Включить bypass mode (если это требуется для данного канала)
2. Заблокировать выходной «ClockGate» (выключить частоту). Блокировка возможна как в автоматическом (с поддержкой Q-Channel интерфейса), так и в ручном режиме.
3. Дождаться подтверждения выключения частоты (Q_FSM_STATE == Q_STOPPED)
4. Записать новый коэффициент деления UCG_CTR*_REG[DIV_COEFF]
5. Дождаться перехода на новый коэффициент деления (в поле DIV_LOCK регистра UCG_CTR*_REG должна появиться 1).
6. Разблокировать выходной «ClockGate» (включить частоту). Разблокировка возможна как в автоматическом (с поддержкой Q-Channel интерфейса), так и в ручном режиме.
7. Дождаться подтверждения включения частоты (Q_FSM_STATE == Q_RUN)
8. Выключить bypass mode (если он был включен)

43.3.5 Bypass mode

Для перевода канала в bypass mode достаточно установить соответствующий бит регистра UCG_BP_CTR_REG в 1.

Важно! Перевод в bypass mode возможен только включенного (генерирующего частоту) канала.

43.3.6 Режимы включения/выключения частоты

В блоке реализована возможность автоматического и ручного управления частотой.

1. Автоматический режим. Включение/выключение частоты производится в соответствии со стандартом “ARM Q-Channel”. Данный режим выбран по умолчанию.

   Запись «1» в поле «CLK_EN» регистра UCG_CTR*_REG инициирует процедуру включения частоты. Текущее значение внутренней машины состояний (Q_FSM)

можно узнать прочитав поле Q_FSM_STATE регистра UCG_CTR*_REG. Процедура включения завершена, если Q_FSM находится в состоянии Q_RUN.

Запись «0» в поле «CLK_EN» регистра UCG_CTR*_REG инициирует процедуру выключения частоты. Процедура выключения завершена, если Q_FSM находится в состоянии Q_STOPPED.

2. Ручной режим. Если нет необходимости (возможности) работать в автоматическом режиме, то предусмотрен ручной режим включения/выключения частоты. В данном режиме не анализируется состояние сигналов Q-Channel интерфейса, а производится прямое управление выходным элементом «ClockGate».

   Запись «1» в поле «CLK_EN» регистра UCG_CTR*_REG приводит к включению частоты. Процедура включения завершена, если Q_FSM находится в состоянии Q_RUN.

   Запись «0» в поле «CLK_EN» регистра UCG_CTR*_REG приводит к выключению частоты. Процедура выключения завершена, если Q_FSM находится в состоянии Q_STOPPED.

ВАЖНО! Переключение режима работы осуществляется битом LPI_EN регистра UCG_CTR*_REG и возможно только при выключенной частоте (когда Q_FSM находится в состоянии Q_STOPPED).

43.3.7 Синхронизация каналов

Под синхронизацией каналов понимается процедура выравнивания фазы сгенерированных частот. Фазы сгенерированных частот могут быть смещены относительно друг друга на неопределенное количество периодов референсной частоты. Это происходит из-за того, что управление каналами (включение, выключение, смена коэффициента деления и т.д.) осуществляется последовательно. Такая разбежка фазы недопустима для некоторых систем.

Выравнивания фазы можно осуществить записью «1» в соответствующие поля регистра «UCG_SYNC_CLK_REG». Например, для выравнивания фазы каналов 0, 3, 7 необходимо записать значение 0x0000_0089.

Процедура синхронизации аналогична процедуре смены коэффициента деления и требует выполнения гейтирования синхронизируемых каналов, либо перевода их в bypass mode.

43.3.8 Описание работы Q_FSM

Внутренняя машина состояний повторяет описанную в документе «IHI0068B_low_power_interface_spec», за исключением того, что добавлено состояние

Q_CLK_EN (между состояниями Q_EXIT и Q_RUN) в течении которого производится аппаратное включение частоты.

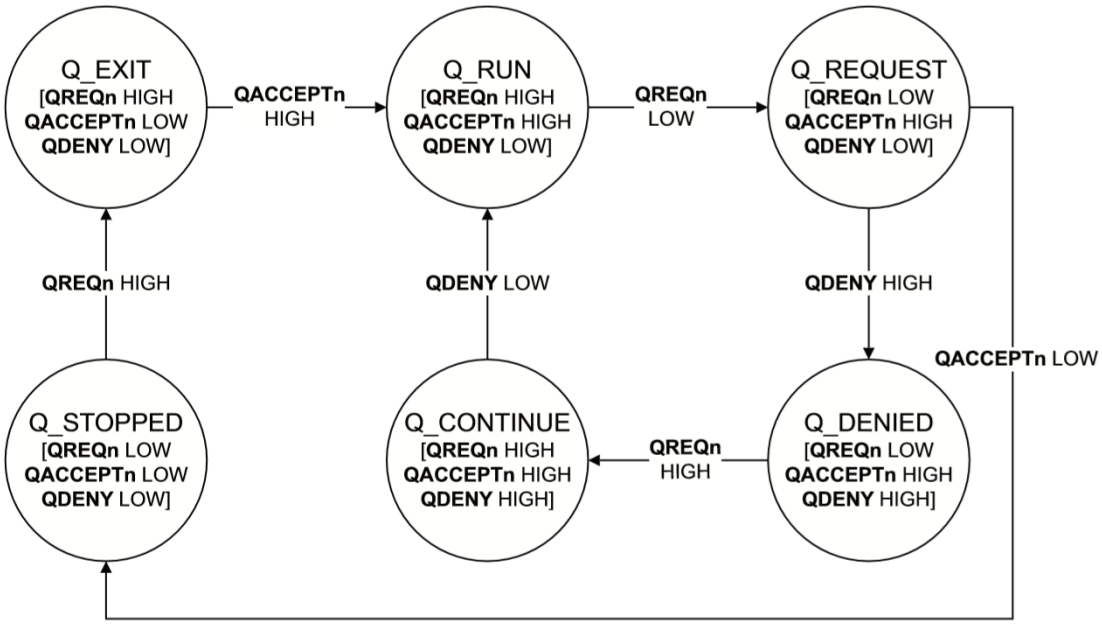


Рисунок 43.3.

44. БЛОК МНОГОПОРТОВОЙ ОБЩЕЙ ПАМЯТИ, ПЕРЕМЕЖИТЕЛЬ (MPSM)

44.1 Общая информация

Блок многопортовой общей памяти (Multi Port Shared Memory, MPSM) содержит память общего назначения, организованную в виде нескольких банков. Со стороны системы доступ ко всему объему памяти возможен по любому из восьми AXI портов.

Также в состав MPSM входит блок перемежения (Interleaver, INTRLV), предназначенный для быстрой перестановки элементов в больших массивах данных. Блок может использоваться в системах связи для выполнения операций перемежения/деперемежения, а также для перфорации помехоустойчивых кодов с целью повышения скорости передачи. Для работы INTRLV используется часть общей памяти MPSM.

Блок MPSM имеет следующие характеристики:

- Общий объем памяти: 4 Мбайта
- Количество AXI портов для доступа к памяти: 8
- Независимые каналы записи и чтения по каждому порту
- Поддерживаются обращения размером 8, 16, 32, 64 и 128 бит, а также невыровненные обращения к памяти
- Количество банков памяти: 8
- Чередование адресов между банками для увеличения скорости доступа
- Round-robin алгоритм выбора порта для предоставления доступа к банку памяти
- Работает на частоте системного интерконнекта
- Характеристики блока INTRLV:
- Формат элемента данных: 1, 8, 16, 32 бит
- Максимальный размер массива данных: 131072 элемента
- Одна таблица индексов для операций перемежения и деперемежения
- Поддержка произвольного расположения начала массивов данных в памяти
- Генерация прерывания при завершении выполнения операции
- Производительность: не менее $50 \cdot 10^6$ перестановок/с

44.2 Структурная схема

На Рисунок 44.1 приведена структурная схема MPSM, на которой показаны следующие элементы:

- Восемь интерфейсов AXI #i (i = 0, 1, … 7), подключенных к коммутатору подсистемы SDR.
- Блоки Port #i для каждого интерфейса AXI #i (i = 0, 1, … 7).
- Конфигурационный интерфейс CFG_AXI для доступа к регистрам.
- Блок перемежения (Interleaver).
- Банки памяти Bank #i RAM (i = 0, 1, … 7).
- Арбитры доступа Bank #i arbiter (i = 0, 1, … 7) для каждого банка памяти.

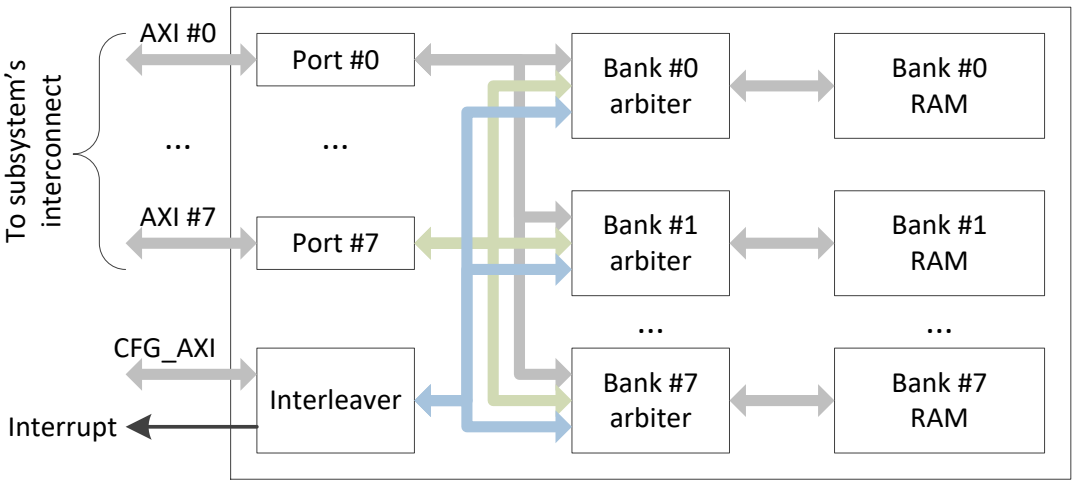


Рисунок 44.1. Структурная схема блока MPSM

44.3 Функционирование

44.3.1 Адресация и доступ к памяти

Общая память MPSM объемом 4 Мбайта доступна по любому из восьми AXI портов в диапазоне смещений адреса 0x0000_0000 – 0x003F_FFFF. Возможен доступ с любым размером слова: 8, 16, 32, 64 или 128 бит. Также поддерживаются невыровненные обращения.

Блоки Port #i (i = 0, 1, … 7) содержат логику для распределения запросов чтения/записи по банкам памяти. Распределение выполняется по чередующимся банкам памяти по 16 байт в каждый банк (Рисунок 44.2).

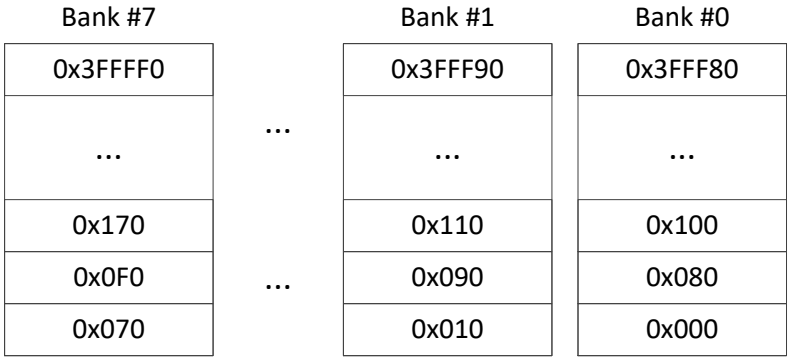


Рисунок 44.2. Распределение запросов по банкам памяти MPSM

Чередование данных по банкам памяти предоставляет возможность одновременного доступа к памяти по нескольким AXI-интерфейсам при условии, что по каждому интерфейсу выполняется чтение/запись по последовательным адресам.

Каждый блок Port #i (i = 0, 1, … 7) подключается к арбитрам банков через два независимых канала – чтения и записи. Перемежитель подключен к арбитрам с помощью трех каналов – канал чтения индексов, канал чтения данных, канал записи данных. Таким образом, каждый арбитр имеет 19 источников запросов к своему банку памяти. Выбор запроса для обработки выполняется за один такт по алгоритму round-robin.

На следующем рисунке показан пример доступа к памяти по нескольким интерфейсам. По интерфейсам AXI #0, AXI #2 и AXI #5 одновременно выполняются последовательные обращения к памяти, начиная с банков #4, #0 и #0 соответственно. Так как запросы по интерфейсам AXI #2 и AXI #5 конфликтуют в начальный момент времени, арбитр предоставляет доступ к банку #0 интерфейсу AXI #2, а на интерфейсе AXI #5 вносится такт ожидания. В последующих тактах запросы от всех интерфейсов распределяются по банкам без конфликтов и обрабатываются одновременно.

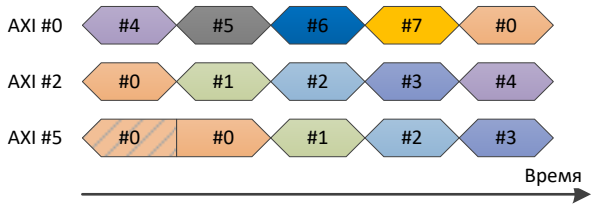


Рисунок 44.3. Пример распределения данных по банкам памяти

44.3.2 Перемежитель

44.3.2.1 Введение

Блок INTRLV работает с тремя массивами, расположенными в общей памяти MPSM:

- массив входных данных X;
- массив выходных данных Y;
- массив индексов P.

Выходной массив Y формируется из элементов X[i] входного массива X путем их перестановки в порядке, задаваемом массивом индексов P.

Размер массива P имеет обозначение N и задается полем BLK_SIZE регистра **MPSM_INTCFG0**. В зависимости от выполняемой операции размеры массивов X и Y могут быть как равны N, так и отличаться от N.

Есть два направления перестановки данных: прямое (перемежение) и обратное (деперемежение). Направление задается битом DIR регистра **MPSM_INTCFG0**.

44.3.2.2 Расположение массивов в памяти

Все три массива P, X, Y располагаются в общей памяти MPSM по адресам, задаваемым в регистрах **MPSM_INTCFG1**, **MPSM_INTCFG2**, **MPSM_INTCFG3** соответственно. Адрес каждого массива должен быть выровнен по 128 бит.

Размер элементов входного и выходного массивов задается параметром DATA_SIZE в регистре **MPSM_INTCFG0** и может принимать следующие значения: 1 бит, 8 бит, 16 бит, 32 бита. В Таблица 44.1– Таблица 44.4 показано расположение элементов массива X в памяти MPSM при различных размерах элементов (для массива Y аналогично).

Таблица 44.1. Расположение элементов массива X в памяти. DATA_SIZE=3 (32 бита)

Разряды слова памяти				Адрес
127:96	95:64	63:32	31:0	
-	X[N-1]	…	X[i]	X_ADDR + floor[(N-1)/4]*16
X[7]	X[6]	X[5]	X[4]	X_ADDR + 0x10
X[3]	X[2]	X[1]	X[0]	X_ADDR + 0x00

Таблица 44.2. Расположение элементов массива X в памяти. DATA_SIZE=2 (16 бит)

Разряды слова памяти								Адрес
127:112	111:96	95:80	79:64	63:48	47:32	31:16	15:0	
-	-	-	-	-	X[N-1]	…	X[i]	X_ADDR + floor[(N-1)/8]*16
X[15]	X[14]	X[13]	X[12]	X[11]	X[10]	X[9]	X[8]	X_ADDR + 0x10
X[7]	X[6]	X[5]	X[4]	X[3]	X[2]	X[1]	X[0]	X_ADDR + 0x00

Таблица 44.3. Расположение элементов массива X в памяти. DATA_SIZE=1 (8 бит)

Разряды слова памяти								Адрес
127:120	119:112	…	…	…	…	15:8	7:0	
-	-	-	-	-	X[N-1]	…	X[i]	X_ADDR + floor[(N-1)/16]*16
X[31]	X[30]	…	…	…	…	X[17]	X[16]	X_ADDR + 0x10
X[15]	X[14]	…	…	…	…	X[1]	X[0]	X_ADDR + 0x00

Таблица 44.4. Расположение элементов массива X в памяти. DATA_SIZE=0 (1 бит)

Разряды слова памяти								Адрес
127	126	…	…	…	…	1	0	
-	-	-	-	-	X[N-1]	…	X[i]	X_ADDR + floor[(N-1)/128]*16
X[255]	X[254]	…	…	…	…	X[129]	X[128]	X_ADDR + 0x10
X[127]	X[126]	…	…	…	…	X[1]	X[0]	X_ADDR + 0x00

Начало массивов X и Y может располагаться в памяти со смещением на несколько элементов относительно выровненного адреса X_ADDR/Y_ADDR. Смещение задается параметрами R_OFFSET и W_OFFSET в регистрах **MPSM_INTCFG4** и **MPSM_INTCFG5** соответственно (см. п. 44.3.2.6).

В Таблица 44.5 показан пример расположения элементов массива X при DATA_SIZE = 0 и R_OFFSET = 2.

Таблица 44.5. Расположение элементов массива X в памяти. DATA_SIZE=0, R_OFFSET=2

Разряды слова памяти								Адрес
127	126	…	…	...	2	1	0	
-	-	-	X[N-1]	…	…	…	X[i]	X_ADDR + floor[(N-1)/128]*16
X[253]	X[252]	…	…	…	…	X[127]	X[126]	X_ADDR + 0x10
X[125]	X[124]	…	…	X[1]	X[0]	-	-	X_ADDR + 0x00

Размер индексного элемента – 17 бит. В Таблица 44.6 показано расположение элементов массива индексов P в памяти MPSM.

Таблица 44.6. Расположение элементов массива P в памяти

Разряды слова памяти								Адрес
127:113	112:96	95:81	80:64	63:49	48:32	31:17	16:0	
-	-	-	-	-	X[N-1]	-	X[i]	P_ADDR + floor[(N-1)/4]*16

Разряды слова памяти								Адрес
127:113	112:96	95:81	80:64	63:49	48:32	31:17	16:0	
-	P[7]	-	P[6]	-	P[5]	-	P[4]	P_ADDR + 0x10
-	P[3]	-	P[2]	-	P[1]	-	P[0]	P_ADDR + 0x00

Примечание: floor[x] – округление до меньшего целого.

44.3.2.3 Операция перемежения

Операция перемежения в общем виде описывается формулой:

$$Y[i + W\_OFFSET] = X[P[i] + R\_OFFSET],\ i = 0, 1, \ldots, (N - 1).$$

Здесь N – размер массива P; X[i], Y[i], P[i] – i-ый элемент соответствующего массива; значения R_OFFSET и W_OFFSET – постоянные смещения.

Адрес i-го элемента считываемого из массива X / записываемого в массив Y вычисляется следующим образом:

$RD\_ADDR_{X[i]} = X\_ADDR + floor((P[i] + R\_OFFSET)*S);$

$WR\_ADDR_{Y[i]} = Y\_ADDR + floor((i + W\_OFFSET)*S);$

где S – размер элемента данных (4, 2, 1 или 1/8 байта).

На Рисунок 44.4 показан пример выполняемой перестановки при N = 5, P = [2, 4, 0, 1, 3], R_OFFSET = 0, W_OFFSET = 0.

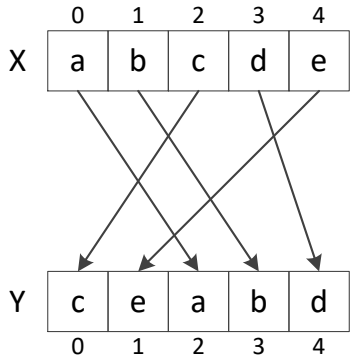


Рисунок 44.4. Пример операции перемежения

44.3.2.4 Операция деперемежения

Операция деперемежения в общем виде описывается формулой:

$$Y[P[i] + W\_OFFSET] = X[i + R\_OFFSET],\ i = 0, 1, \ldots, (N - 1).$$

Здесь N – размер массива P; X[i], Y[i], P[i] – i-ый элемент соответствующего массива; значения R_OFFSET и W_OFFSET – постоянные смещения.

Адрес i-го элемента считываемого из массива X / записываемого в массив Y вычисляется следующим образом:

$RD\_ADDR_{X[i]}$ = X_ADDR + floor((i + R_OFFSET)*S);

$WR\_ADDR_{Y[i]}$ = Y_ADDR + floor((P[i] + W_OFFSET)*S);

где S – размер элемента данных (4, 2, 1 или 1/8 байта).

На Рисунок 44.5 показан пример выполняемой перестановки при N = 5, P = [2, 4, 0, 1, 3], R_OFFSET = 0, W_OFFSET = 0.

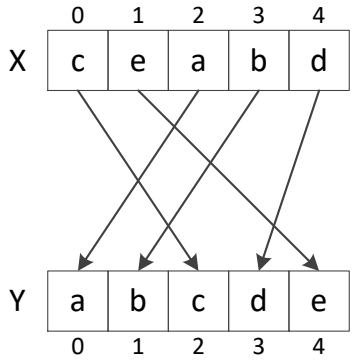


Рисунок 44.5. Пример операции деперемежения

44.3.2.5 Операция перфорации

Перфорация является частным случаем перемежения. При этом количество элементов в выходном массиве Y может быть меньше, чем количество элементов во входном массиве X.

На Рисунок 44.6 показан пример перфорации при N = 3, P = [0, 2, 3], R_OFFSET = 0, W_OFFSET = 0.

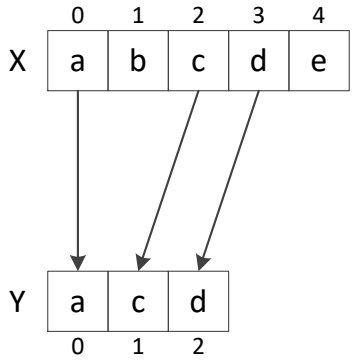


Рисунок 44.6. Пример операции перфорации

Для выполнения обратной перфорации может быть использована операция деперемежения. При этом размер выходного массива Y больше, чем размер входного массива X

На Рисунок 44.7 показан пример обратной перфорации при N = 5, P = [0, 2, 3, 5, 7], R_OFFSET = 0, W_OFFSET = 0. Предполагается, что в начальный момент массив Y заполнен нулями.

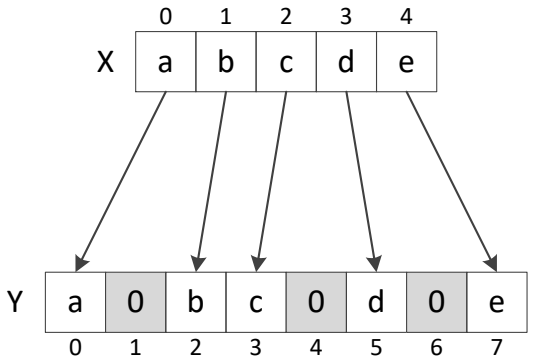


Рисунок 44.7. Пример операции обратной перфорации

44.3.2.6 Использование смещений R_OFFSET, W_OFFSET

Дополнительные смещения индекса могут быть использованы, когда начало массива X или Y в памяти смещено на несколько элементов относительно адреса X/Y_ADDR.

На Рисунок 44.8 показан пример перфорации с использованием смещения W_OFFSET при N = 3, P = [0, 2, 3]. В данном примере требуется выполнить перфорацию двух массивов X_0 и X_1. При этом элементы выходных массивов Y_0 и Y_1 должны располагаться непрерывно в памяти. Для этого перфорация массива X_0 выполняется с W_OFFSET = 0, а массива X_1 – с W_OFFSET=3. При этом в обоих случаях используется один и тот же адрес Y_ADDR массива Y_0, выровненный по 128 бит.

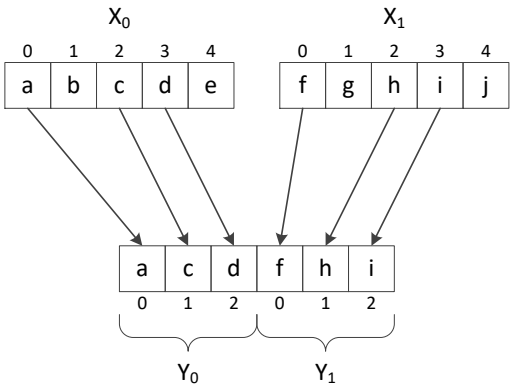


Рисунок 44.8. Пример операции перфорации со смещением

44.3.2.7 Запуск перемежителя

Для запуска перемежителя необходимо:

1. Записать массивы входных данных и индексов в память MPSM по любому интерфейсу AXI.

2. Записать настройки перемежителя и адреса массивов в регистры **MPSM_INTCFG*** по конфигурационному интерфейсу CFG_AXI.

3. Разрешить прерывание INTRLV_DONE в регистре **MPSM_IRQ_EN**.

4. Записать значение 0x1 в регистр **MPSM_INTCMD**.

По завершению работы будет сгенерировано прерывание INTRLV_DONE. Для очистки прерывания необходимо записать значение 0x1 в регистр **MPSM_IRQ_CLR**.

44.3.2.8 Программный сброс

Для выполнения программного сброса перемежителя необходимо записать значение 0x3 в регистр **MPSM_INTCMD**.

Программный сброс останавливает запущенную операцию и приводит перемежитель в исходное состояние. Также сбрасываются регистры **MPSM_INTSTAT**, **MPSM_IRQ_EN**, **MPSM_IRQ_STAT**. Состояние регистров **MPSM_INTCFG*** не изменяется.

44.4 Регистры

44.4.1 Перечень регистров

Ниже приведена сводная таблица регистров блока MPSM, доступных по конфигурационному интерфейсу CFG_AXI. Возможен доступ 32-х, 64-х или 128-ми разрядными словами.

Таблица 44.7. Перечень конфигурационных регистров MPSM

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
MPSM_REVID	Версия RTL	0x19[1]	RO	0x00
MPSM_INTSTAT	Регистр статуса блока перемежения	0x0	RO	0x04
MPSM_INTCFG0	Регистр 0 настройки блока перемежения	0x0	RW	0x08
MPSM_INTCFG1	Регистр 1 настройки блока перемежения	0x0	RW	0x0C
MPSM_INTCFG2	Регистр 2 настройки блока перемежения	0x0	RW	0x10
MPSM_INTCFG3	Регистр 3 настройки блока перемежения	0x0	RW	0x14
MPSM_INTCFG4	Регистр 4 настройки блока перемежения	0x0	RW	0x18
MPSM_INTCFG5	Регистр 5 настройки блока перемежения	0x0	RW	0x1C
MPSM_INTCMD	Командный регистр перемежителя	0x0	WO	0x20
MPSM_IRQ_EN	Регистр разрешения прерываний	0x0	RW	0x30
MPSM_IRQ_STAT	Регистр статуса прерываний	0x0	RW	0x34
MPSM_IRQ_EN_STAT	Регистр статуса разрешенных прерываний	0x0	RO	0x38
MPSM_IRQ_CLR	Регистр сброса прерываний	0x0	WO	0x3C

Примечания:
1) Версия RTL имеет формат 0xXY и соответствует версии SVN релиза. Например, для релиза r1p2 X = 1, Y = 2, значение регистра – 0x12

44.4.2 Регистр MPSM_REVID

Регистр хранит версию RTL в формате rXpY.

Таблица 44.8. Формат регистра MPSM_REVID

Номер бита	Условное обозначение	Назначение
31:8	-	Резерв
7:4	X	Старшая часть номера версии (X)
3:0	Y	Младшая часть номера версии (Y)

44.4.3 Регистр MPSM_INTSTAT

Регистр статуса перемежителя.

Таблица 44.9. Формат регистра MPSM_INTSTAT

Номер бита	Условное обозначение	Назначение
31:1	-	Резерв
0	RUN	Статус работы блока перемежения: 0: перемежитель не запущен; 1: перемежитель запущен

44.4.4 Регистр MPSM_INTCFG0

Таблица 44.10. Формат регистра MPSM_INTCFG0

Номер бита	Условное обозначение	Назначение
31:25	-	Резерв
24:8	BLK_SIZE	Количество элементов в массиве P минус 1 (N – 1)
7:3	-	Резерв
2	DIR	Направление перемежения: 0: прямое (перемежение); 1: обратное (деперемежение)
1:0	DATA_SIZE	Размер элемента данных: 0: 1 бит; 1: 8 бит; 2: 16 бит; 3: 32 бита

44.4.5 Регистр MPSM_ INTCFG1

Таблица 44.11. Формат регистра MPSM_INTCFG1

Номер бита	Условное обозначение	Назначение
31:22	-	Резерв
21:0	P_ADDR	Адрес массива индексов P в памяти MPSM

44.4.6 Регистр MPSM_INTCFG2

Таблица 44.12. Формат регистра MPSM_INTCFG2

Номер бита	Условное обозначение	Назначение
31:22	-	Резерв
21:0	X_ADDR	Адрес массива входных данных X в памяти MPSM

44.4.7 Регистр MPSM_INTCFG3

Таблица 44.13. Формат регистра MPSM_INTCFG3

Номер бита	Условное обозначение	Назначение
31:22	-	Резерв
21:0	Y_ADDR	Адрес массива выходных данных Y в памяти MPSM

44.4.8 Регистр MPSM_INTCFG4

Таблица 44.14. Формат регистра MPSM_INTCFG4

Номер бита	Условное обозначение	Назначение
31:17	-	Резерв
16:0	R_OFFSET	Смещение, прибавляемое к индексу при чтении

44.4.9 Регистр MPSM_INTCFG5

Таблица 44.15. Формат регистра MPSM_INTCFG5

Номер бита	Условное обозначение	Назначение
31:17	-	Резерв
16:0	W_OFFSET	Смещение, прибавляемое к индексу при записи

44.4.10 Регистр MPSM_INTCMD

Таблица 44.16. Формат регистра MPSM_INTCMD

Номер бита	Условное обозначение	Назначение
31:2	-	Резерв
1:0	CMD	Выполняемая команда: 0: резерв; 1: запуск перемежения; 2: резерв; 3: остановка и сброс блока перемежения в начальное состояние

44.4.11 Регистр MPSM_IRQ_EN

Таблица 44.17. Формат регистра MPSM_IRQ_EN

Номер бита	Условное обозначение	Назначение
31:1	-	Резерв
0	INTRLV_DONE	Разрешение прерывания по завершению операции перемежения

44.4.12 Регистр MPSM_IRQ_STAT

Регистр содержит биты статуса для всех прерываний. Бит статуса устанавливается в 1 при возникновении прерывания без учета разрешений, либо при записи 1 в этот бит (эта возможность может быть использована для отладочных целей). Запись 0 игнорируется.

Таблица 44.18. Формат регистра MPSM_IRQ_STAT

Номер бита	Условное обозначение	Назначение
31:1	-	Резерв
0	INTRLV_DONE	Статус прерывания по завершению операции перемежения

44.4.13 Регистр MPSM_IRQ_EN_STAT

Регистр содержит биты статуса для всех разрешенных прерываний. Бит статуса устанавливается в 1 при возникновении прерывания только, если установлен соответствующий бит в регистре **MPSM_IRQ_EN**. Регистр доступен только для чтения.

Таблица 44.19. Формат регистра MPSM_IRQ_EN_STAT

Номер бита	Условное обозначение	Назначение
31:1	-	Резерв
0	INTRLV_DONE	Статус прерывания по завершению операции перемежения

44.4.14 Регистр MPSM_IRQ_CLR

Регистр предназначен для программного сброса прерываний, бит статуса которых установлен в регистре **MPSM_IRQ_STAT**. Сброс производится путем записи 1 в соответствующий бит. Запись 0 игнорируется. При чтении регистра возвращается 0.

Таблица 44.20. Формат регистра MPSM_IRQ_CLR

Номер бита	Условное обозначение	Назначение
31:1	-	Резерв
0	INTRLV_DONE	Сброс прерывания по завершению операции перемежения

45. УНИВЕРСАЛЬНЫЙ БЛОК ФОРМИРОВАНИЯ СБРОСА (URG)

Блок УБФС позволяет производить сброс необходимых блоков (далее локальных) в нужной последовательности.



Рисунок 45.1. Верхний уровень блока УБФС

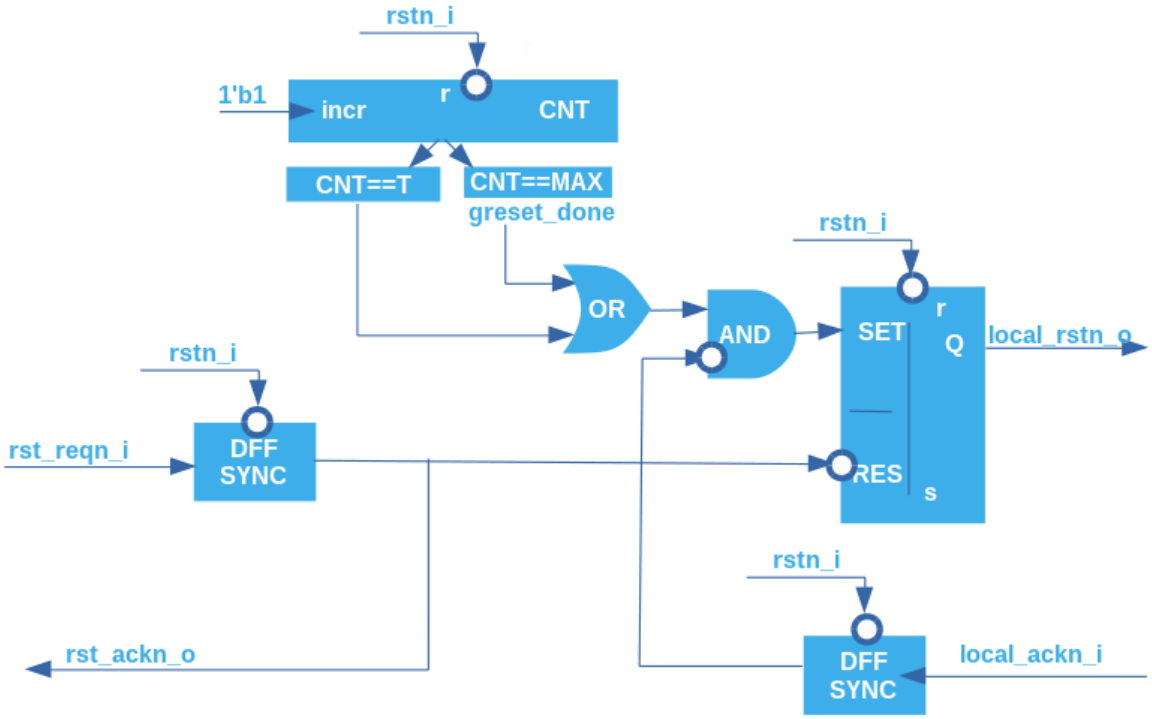


Рисунок 45.2. Детали УБФС

Условиями установки уровня «0» на выходе local_rstn_o[k] являются:

- установка входа rstn_i в «0» (асинхронный сброс)
- установка rst_reqn_i[k] в «0» (синхронный сброс)

Для того, чтобы local_rstn_o[k] после установки "0" стал единицей необходимо одновременное выполнение следующий условий:

- на входе local_ackn_i[k] должен быть установлен и удерживаться «0»
- сигнал rst_reqn_i[k] должен быть установлен и удерживаться в «1»
- все блоки сброшены после снятия rstn_i (счётчик времени сброса достиг максимального значения) или счётчик времени достиг значения TIME_TO_RISE[k].

45.1 Полный сброс по rstn_i

При установке активного уровня rstn_i все сигналы local_rstn_o[k] немедленно переходят в активное состояние (акт. 0). После снятия rstn_i сигнал local_rstn_o[k] переводится в неактивное состояние через TIME_TO_RISE[k] тактов clk (если сигнал local_ackn_i уст. в "0" и не установлен сигнал rst_reqn_i[k]).

Время TIME_TO_RISE[k] - параметр передаваемый блоку.

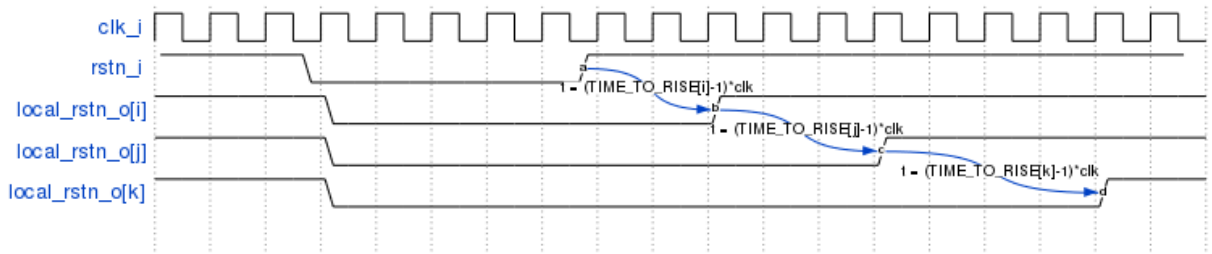


Рисунок 45.3.

45.2 Сброс выбранного блока или набора блоков (локальный сброс)

Для того, чтобы сбросить необходимой блок, нужно установить сигнал rst_reqn_i[k] в «0» и удерживать его до появления «0» на выходе rst_ackn_o[k] (это будет является признаком того, что сигнал rst_reqn_i[k] защелкнут на внутреннем регистре блока УБФС).

local_ackn_i[k] - Признак того, что блок УБФС может снять акт. уровень local_rstn_o[k]. Пока этот вход установлен в «1» сигнал local_rstn_o[k] не снимется. Если не нужно особенного управления сбросом блока, этот вход можно подключить к «0».

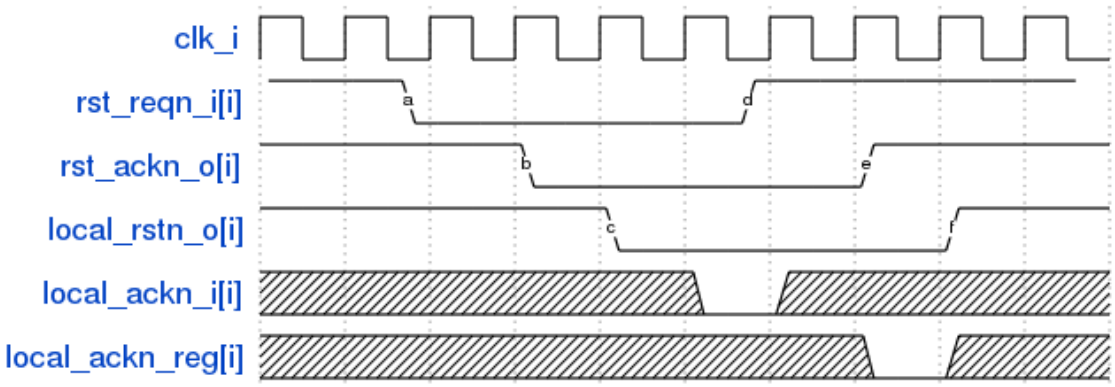


Рисунок 45.4.

(на диаграмме показан сигнал local_ackn_i_reg – значение local_ackn_i после пересинхронизации)

45.3 Порты и параметры

Все входы блока – асинхронны.

Таблица 45.1.

Порт	Направление	Комментарии
clk	Вход	Рабочая частота блока
rstn_i	Вход	Асинхронный глобальный сброс с активным '0'. Должнен быть подсинхронизован под clk снаружи блока УБФС. При установке rstn_i в 0 сигналы local_rstn_o[k] асинхронно устанавливаются в '0'. После снятия rstn_i сигналы local_rstn_o[k] устанавливаются в единицу согласно настройкам.
local_rstn_o[k]	Выход	Сброс генерируемый блоком (локальный)
local_ackn_i[k]	Вход	Признак того, что сброс начат (сигнал local_rstn_o защелкнут на внутреннем регистре УБФС)
rst_reqn_i[k]	Вход	Запрос установки локального сброса
rst_ackn_o[k]	Выход	Подтверждение окончания локального сброса

Таблица 45.2.

Параметр	Значение по умолчанию	Описание
LOCAL_RESETS_NUM	4	Количество локальных сбросов.
TIME_TO_RISE	{4'h0,4'h1,4'h2,4'h3}	Шина параметров - задаёт момент выхода из ресета для соответствующего локального сброса. Например, {T0, T1,T2, T3} - конкантенация всех параметров для каждого локального порта .
TIME_WIDTH	4	Разрядность счётчика для отсчёта времени TIME_TO_RISE.

TIME_MAX	$2^{TIME\_WIDTH}-1$	Значение времени, после которого считается, что глобальный сброс завершён.После чего начинают обрабатываться запросы local_ackn_i. **Важно, чтобы этот параметр был равен макимальному значению TIME_TO_RISE.**

46. МНОГОКАНАЛЬНОЕ DMA ПАМЯТЬ-ПАМЯТЬ (QDMA)

46.1 Общая информация

QDMA реализует 4 независимых DMA канала память-память, с возможностью двумерной адресации и функциями самоинициализации.

46.1.1 Типы каналов

Контроллер QDMA имеет 4 канала типа память-память. Перечень каналов приведен в Таблица 46.1.

Таблица 46.1. Каналы DMA

Условное обозначение канала	Назначение канала
MEM_CH0 – MEM_CH3	Обмен данными типа память-память

Памятью могут как внутренняя память СнК, так внешняя память, доступная через порты памяти СнК.

Если при работе QDMA изменяется программный код RISC в памяти, то когерентность кэш программ RISC (ICACHE), аппаратно не обеспечивается. В этом случае для обеспечения когерентности используется бит FLUSH в регистре CSR RISC.

46.1.2 Темп передачи

QDMA осуществляют передачу 64/32-разрядными словами данных.

Каналы за один цикл занятия коммутатора передают пачку данных. Размер пачки задается полем WN в регистре CSR соответствующего канала DMA и определяется системными требованиями по передаче данных. Если после передачи пачки данных нет запросов от других каналов QDMA или VCPU, то данный канал без перерыва начинает передавать следующую пачку данных и т.д.

46.1.3 Прерывания DMA

Канал DMA формирует прерывание)

- при единичном состоянии бита DONE;
- при единичном состоянии битов END.

Обнуление битов DONE и END (и снятие соответствующего прерывания) выполняется посредством чтения содержимого регистра CSR или записью в них нуля.

46.2 Процедура самоинициализации

Каналы QDMA MEM_CH могут выполнять процедуру самоинициализации (выполнение цепочки передач DMA).

Для выполнения самоинициализации в каналах имеется 32-разрядный регистр CP, в котором хранится начальный адрес блока параметров очередного DMA обмена. Младшие три разряда регистра CP игнорируются (адреса выровнены по границе 64-разрядного слова). Младший (нулевой разряд) регистра CP используется для старта режима самоинициализации. Эти параметры при самоинициализации аппаратно загружаются в 64-разрядном формате в соответствующие регистры канала DMA. Процедура этой загрузки ничем не отличается от обычного DMA обмена. Блок параметров может размещаться в любой памяти.

Параметры для самоинициализации размещаются в памяти в шести последовательных 32-разрядных словах, следующим образом (в порядке возрастания адресов):

Таблица 46.2. Параметры для самоинициализации

Смещение	Параметр
0x00	IR1
0x04	IR0
0x08	{WCY_{16}, ORY_{16}}
0x0C	{$OR1_{16}$,$OR0_{16}$}
0x10	CSR
0x14	CP

Если необходимо продолжить цепочку команд, то необходимо указать CHEN=1. В режиме самоинициализации при записи параметров в регистр CSR биты END и DONE недоступны.

Для запуска работы канала DMA в режиме с самоинициализацией необходимо в регистр CP записать адрес первого блока параметров DMA передачи. При этом 0 разряд записываемых данных должен содержать 1 (признак пуска самоинициализации). В результате этого, соответствующий канал загрузит в свои регистры параметры DMA передачи и начнет обмен данными.

После окончания передачи блока данных бит END в регистре CSR устанавливается в единичное состояние, если бит IM = 1 - выдается прерывание. По окончании передачи блока данных также проверяется состояние бита CHEN. Если он равен 1, то будет загружен следующий блок параметров DMA передачи и т.д. В противном случае цепочка DMA

обменов закончится и в регистре CSR бит DONE установится в единичное состояние и выдается прерывание.

При необходимости каналы DMA могут инициализироваться программно. Для этого должны быть проинициализированы все необходимые регистры индекса и смещения, а затем регистр CSR. При загрузке регистра CSR бит RUN необходимо установить в единичное состояние. Следует отметить, что бит RUN может быть использован для приостановки канала DMA. Для этого в любой момент времени в него необходимо записать 0. Для продолжения работы соответственно в бит RUN необходимо записать 1. Бит RUN может быть использован также для приостановки выполнения цепочки, если при загрузке очередных параметров он будет равен 0. Для продолжения выполнения цепочки в бит RUN необходимо записать 1. Для удобства организации обмена только с битом RUN выделен персональный адрес в адресном пространстве канала DMA MEM_CH.

Параметры для самоинициализации размещаются в памяти в двух последовательных 64-разрядных словах, следующим образом (в порядке возрастания адресов):

Смещение	Параметр
0x00	IR
0x04	-
0x08	CSR
0x0C	CP

46.3 Программная модель QDMA

В QDMA имеется 4 канала MEM_CH, которые обеспечивают обмен данными между двумя областями любых блоков памяти (внутренних или внешних).

Для управления работой каждого канала MEM_CH имеются следующие регистры:

- регистр управления и состояния – CSR_MEM_CH;
- регистры индекса (физический адрес памяти) - IR0, IR1;
- регистры смещения - OR, Y;
- регистр начального физического адреса блока параметров DMA передачи для выполнения процедуры самоинициализации (CP);
- псевдорегистр управления состоянием бита RUN регистра CSR (RUN_MEM_CH).

Исходное состояние регистров CSR_MEM_CH: разряды [15:0] – нули, а состояние разрядов 31:16 не определено. Исходное состояние остальных регистров не определено.

Карта памяти QDMA и смещение регистров управления каналами приведены в

Таблица 46.3. Регистры QDMA

Смещение	Название	Описание
Канал 0		
0x0000	CSR_MemCh0	Регистр управления и состояния канала MemCh0

Смещение	Название	Описание
0x0004	IOR_MemCh0	Регистр индекса и смещения внутренней памяти канала MemCh0
0x0008	CP_MemCh0	Регистр указателя цепочки канала MemCh0
0x000C	IR_MemCh0	Индексный регистр внешней памяти канала Memh0
0x0010	OR_MemCh0	Регистр смещения внешней памяти канала MemCh0
0x0014	Y_MemCh0	Регистр параметров направления Y при двухмерной адресации внешней памяти канала MemCh0
0x0018	Run0	Псевдорегистр управления состоянием бита RUN регистра CSR_MemCh0
Канал 1		
0x0020	CSR_MemCh1	Регистр управления и состояния канала MemCh1
0x0024	IOR_MemCh1	Регистр индекса и смещения внутренней памяти канала MemCh1
0x0028	CP_MemCh1	Регистр указателя цепочки канала MemCh1
0x002C	IR_MemCh1	Индексный регистр внешней памяти канала MemCh1
0x0030	OR_MemCh1	Регистр смещения внешней памяти канала MemCh1
0x0034	Y_MemCh1	Регистр параметров направления Y при двухмерной адресации внешней памяти канала MemCh1
0x0038	Run1	Псевдорегистр управления состоянием бита RUN регистра CSR_MemCh1
Канал 2		
0x0040	CSR_MemCh2	Регистр управления и состояния канала MemCh2
0x0044	IOR_MemCh2	Регистр индекса и смещения внутренней памяти канала MemCh2
0x0048	CP_MemCh2	Регистр указателя цепочки канала MemCh2
0x004C	IR_MemCh2	Индексный регистр внешней памяти канала Memh2
0x0050	OR_MemCh2	Регистр смещения внешней памяти канала MemCh2
0x0054	Y_MemCh2	Регистр параметров направления Y при двухмерной адресации внешней памяти канала MemCh2
0x0058	Run2	Псевдорегистр управления состоянием бита RUN регистра CSR_MemCh2
Канал 3		
0x0060	CSR_MemCh3	Регистр управления и состояния канала MemCh3
0x0064	IOR_MemCh3	Регистр индекса и смещения внутренней памяти канала MemCh3
0x0068	CP_MemCh3	Регистр указателя цепочки канала MemCh3
0x006C	IR_MemCh3	Индексный регистр внешней памяти канала Memh3
0x0070	OR_MemCh3	Регистр смещения внешней памяти канала MemCh3
0x0074	Y_MemCh3	Регистр параметров направления Y при двухмерной адресации внешней памяти канала MemCh3
0x0078	Run3	Псевдорегистр управления состоянием бита RUN регистра CSR_MemCh3

Формат регистров CSR_MEM_CH этих каналов приведен в Таблица 46.4.

Таблица 46.4. Формат регистра управления и состояния каналов MEM_CH

Номер разряда	Условное Обозначение	Назначение
0	RUN	Состояние работы канала DMA: 0 – состояние останова; 1 – состояние обмена данными. Устанавливается в 1 при записи 1 в этот разряд. Устанавливается в 0: при записи 0 в этот разряд и после окончания передачи данных, оставшихся в канале; при завершении передачи блока данных.

Номер разряда	Условное Обозначение	Назначение
		Состояние этого бита определяется в процессе выполнения процедуры самоинициализации
1	DIR	Направление обмена данными: 0 – память по IR0 => память по IR1; 1 – память по IR1 => память по IR0.
5:2	WN	Пакет данных, который передается по коммутатору AXI Switch за одно предоставление прямого доступа: 0 – 1 слово; F – 16 слов. Посредством этого параметра можно плавно изменять приоритет каналов DMA относительно CPU, DSP и относительно друг друга
6	EN64	Формат передаваемых данных по коммутатору AXI Switch: 0 – 32 разряда; 1 – 64 разряда. При передаче 32-разрядными словами: WCX – число 32-разрядных слов; адрес в IR0, IR1 должен быть выровнен по границе 32-разрядного слова. При передаче 64-разрядными словами: WCX – число 64-разрядных слов; адрес в IR0, IR1 должен быть выровнен по границе 64-разрядного слова
7	START_DSP	Разрешение запуска работы DSP-ядра (перевод из состояния STOP в состояние RUN) после завершения передачи блока данных: 0 – запуск запрещен; 1 – запуск разрешен.
8	MODE	Режим модификации адреса регистра IR0 0 – линейный режим; 1 – режим с обратным переносом.
9	2D	Режим модификации адреса регистра IR1: 0 – одномерный режим; 1 – двухмерный режим.
10	MASK	Маска внешнего запроса прямого доступа nDMAR: 0 – запрос запрещен; 1 – запрос разрешен. Если разряд равен нулю, то канал работает только под управлением бита RUN. Если разряд равен 1, то для инициализации канала необходимо также наличие запроса nDMAR (низкий уровень).
11	FLYBY	Признак выполнения обмена данными в режиме Flyby: 0 – обычный режим; 1 – режим Flyby. Обмен данными между внешней памятью и внешним устройством
12	CHEN	Разрешение выполнения очередной процедуры самоинициализации: 0 – выполнение очередной процедуры самоинициализации запрещено; 1 – выполнение очередной процедуры самоинициализации разрешено. Используется только при обмене цепочкой блоков данных. Состояние этого бита определяется в процессе выполнения процедуры самоинициализации
13	IM	Маска разрешение установки признака END: 0 – установки признака запрещено; 1 – установки признака разрешено.

Номер разряда	Условное Обозначение	Назначение
		Используется только при обмене цепочкой блоков данных. Состояние этого бита определяется в процессе выполнения процедуры самоинициализации
14	END	Признак завершения передачи блока данных. Устанавливается в 1 при завершении передачи блока данных (при IM=1). Устанавливается в 0 при чтении содержимого этого регистра. Доступен по записи и чтению.
15	DONE	Признак завершения передачи блока данных. Устанавливается в 1 при завершении передачи блока данных при CHEN=0 (CHEN=1 может быть только при использовании процедуры самоинициализации). Устанавливается в 0 при чтении содержимого этого регистра
31:16	WCX	Число слов данных, которые должен передать канал DMA при одномерной адресации (блок данных). Число слов в строке при двухмерной адресации. Количество передаваемых слов = WCX + 1. Содержимое этого поля уменьшается на 1 после передачи каналом DMA очередного слова данных

Все разряды регистра CSR_MEM_CH доступны по записи и чтению.

Состоянием разряда 0 регистра CSR_MEM_CH можно управлять, используя адрес псевдорегистра RUN. При этом остальные разряды этого регистра не изменяются. Эта процедура может быть использована для временной приостановки канала DMA. При чтении по адресу псевдорегистра RUN считывается содержимое регистра CSR_MEM_CH без сброса битов END и DONE.

32-разрядные регистры индекса IR0, IR1 содержат начальные физические адреса источника и приемника данных (или, наоборот, в зависимости от содержимого разряда DIR регистра CSR_MEM_CH) памяти микросхемы. В зависимости от содержимого разряда EN64 адреса в этих регистрах должны быть выровнены по границе 32 или 64-разрядного слова.

Формат регистра смещения OR приведен в Таблица 46.5.

Таблица 46.5. Формат регистра индекса и смещения каналов MEM_CH

Номер разряда	Условное обозначение	Назначение
15:0	OR0	Смещение (приращение) адреса для индексного регистра IR0 после передачи каждого слова данных
31:16	OR1	Смещение (приращение) адреса для индексного регистра IR1 после передачи каждого слова данных

Модификация индексного регистра IR0 при помощи смещения OR0 обеспечивается в режимах с прямым или обратным переносами. Режим с обратным переносом используется при реализации алгоритма быстрого преобразования Фурье (БПФ). Модификация индексного регистра IR1 при помощи смещения OR1 обеспечивается только в режиме с прямым переносом.

В режиме модификации индексного регистра с прямым переносом смещение, задаваемое полями OR0, OR1, рассматривается как число со знаком в диапазоне –32768 до +32767 слов данных (32 или 64-разрядных). Алгоритм модификации адреса с прямым переносом:

for (x = 0; x < WCX; x++) { пересылка по адресу IR0;

модификация адреса для 64-х разрядного обмена: IR0 = IR0 + {{13{OR0[15]}},OR0,000};

модификация адреса для 32-х разрядного обмена: IR0 = IR0 + {{14{OR0[15]}},OR0,00};

пересылка по адресу IR1;

модификация адреса для 64-х разрядного обмена: IR1 = IR1 + {{13{OR1[15]}},OR1,000};

модификация адреса для 32-х разрядного обмена: IR1 = IR1 + {{14{OR1[15]}},OR1,00};

}

В режиме модификации индексного регистра с обратным переносом смещение, задаваемое полем OR0, имеет диапазон от 0 до +65535. Модификация адреса в этом случае выполняется с распространением переноса в обратном направлении – от старших разрядов к младшим. Операция модификации адреса с обратным переносом эквивалентна последовательному выполнению следующих процедур:

- 16-разрядное смещение OR0 дополняется до 32 разрядов: со стороны младших – двумя или тремя нулями, (для 32 или 64-разрядного обменов соответственно, а со стороны старших разрядов – четырнадцатью или тринадцатью нулями, для 32 и 64-разрядного обменов соответственно;
- изменение на обратный порядок следования разрядов в регистрах адреса и смещения. При этом старший бит становится младшим;
- модификация адреса посредством операции сложения с прямым переносом;
- восстановление первоначального порядка следования разрядов регистра адреса.

Модификацию адреса с обратным переносом можно описать при помощи следующих выражений:

IR0 [0:31] = IR0[0:31] + {000, OR0[0:15], 0000000000000} – для 64-разрядного обмена;

IR0 [0:31] = IR0[0:31] + {00, OR0[0:15],00000000000000} – для 32-разрядного обмена.

Каналя MEM_CH обеспечивают передачу двумерных массивов (матриц W[m;n]). При этом, память (внутренняя или внешняя) адресуется в двухмерном режиме. Для этого имеется 32-разрядный регистр Y, формат которого приведен в Таблица 46.6.

Таблица 46.6. Формат регистра Y

Номер разряда	Условное обозначение	Назначение
15:0	OY	Смещение (приращение) адреса памяти в 32-разрядных словах по направлению Y. Используется только при двухмерной адресации.
31:16	WCY	Число строк по Y направлению. Используется только при двухмерной адресации. Количество передаваемых строк = WCY + 1.

При двухмерном режиме адресации поле WCX регистра CSR содержит число слов в строке (X направление), а поле WCY регистра Y содержит число строк (Y направление). Пересылка каждого слова данных осуществляется по индексному регистру IR1 с его последующей инкрементацией на величину, соответствующую содержимому поля OR1 регистра OR (X направление) или поля OY регистра Y. Двухмерная адресация выполняется следующим образом:

Содержимое счетчика WCX сохраняется в буферном регистре;

1 цикл. Индексный регистр внешней памяти модифицируется с использованием смещения OR1. Счетчик WCX декрементируется. Если он равен 0, то переход ко второму циклу.

2 цикл. Состояние счетчика WCX восстанавливается из буферного регистра. Индексный регистр внешней памяти модифицируется с использованием смещения OY. Счетчик WCY декрементируется. Если он не равен 0, то переход к первому циклу. Если он равен 0, то работа канала завершается.

Функционально двумерная адресация эквивалентна следующему двойному циклу (реализуется только по IR1, OR1):

```
for ( y = 0; y <= WCY; y++ ) {
                                for ( x = 0; x < WCX; x++ ) { пересылка по адресу IR1
                                                              для 64-х разрядного обмена : IR1 =
IR1 + {{13{OR1[15]}},OR1,3'h0};
                                                              для 32-х разрядного обмена : IR1 =
IR1 + {{14{OR1[15]}},OR1,2'h0}
                                                             };
                                пересылка по адресу IR1
                                для 64-х разрядного обмена : IR1 = IR1 +
{{13{ORY[15]}},ORY,3'h0};
                                для 32-х разрядного обмена : IR1 = IR1 +
{{14{ORY[15]}},ORY,2'h0};
                               };

//общее кол-во пересылок (WCX=1)*(WCY+1)
```

47. УНИВЕРСАЛЬНОЕ DMA ПЕРИФЕРИЙНЫХ УСТРОЙСТВ (PDMA)

47.1 Общая информация

PDMA - устройство шины AHB, позволяющее автоматически (без постоянного вмешательства CPU) выполнять пересылки данных между заданными устройствами шины. Структурная схема PDMA и пример работы показаны на Рисунок 47.1:

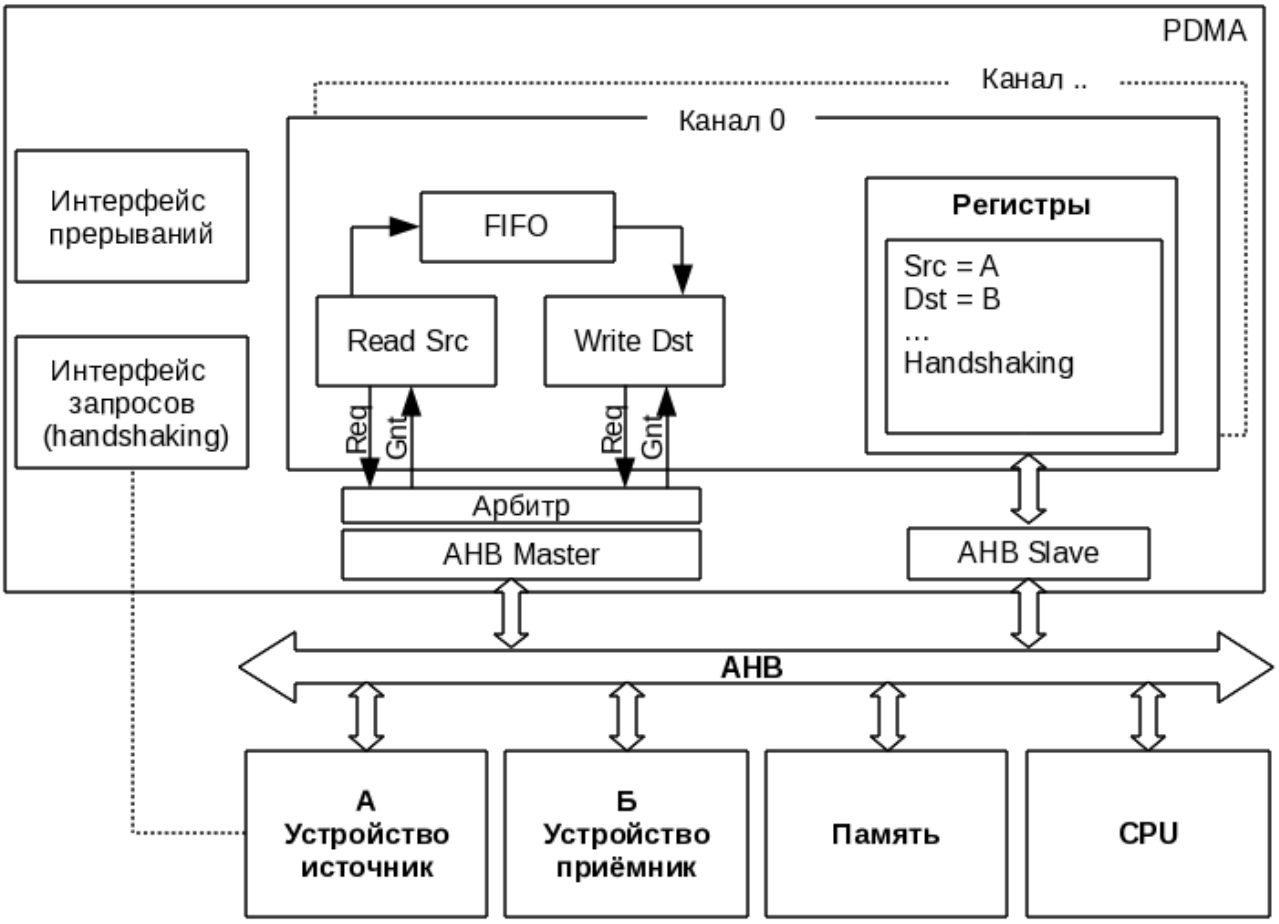


Рисунок 47.1. Структура PDMA и пример работы

PDMA читает данные из *Источника* А, сохраняет их в FIFO, затем берет данные из FIFO и записывает их в *Приёмник* Б.

Источник - устройство на шине AHB, из которого PDMA читает данные. *Приёмник* - устройство на шине AHB, в которое PDMA производит запись данных.

В качестве *Источника*/*Приёмника* может выступать любое устройство, доступное с данной шины AHB. Например, ведомое устройство шины AHB или APB. В случае APB доступ к источнику производится через мост AHB-APB.

Источник и приёмник образуют *Канал пересылки* (далее *Канал*). Всего PDMA позволяет настроить до 8-ми одновременно работающих *Каналов*. Каждый *Канал* имеет свой собственный набор регистров управления, отдельное, собственное FIFO.

Пересылка данных осуществляется через ведущий порт AHB.

Арбитр ведущего порта AHB PDMA управляет доступом *Каналов* к шине на основе их приоритета (настраивается программно).

Настройка и управление пересылками осуществляется через ведомый порт AHB. Например, задаётся адрес *Источника*, адрес *Приёмника*, размер пересылки и др.

Инициирование передач непосредственно на шине AHB осуществляется через *Интерфейс запросов* (программный или аппаратный). При использовании аппаратного *Интерфейса запросов* периферийное устройство (*Источник/Приёмник*) устанавливает аппаратный сигнал готовности к передаче, после чего PDMA запускает передачу на шине AHB. При использовании программного *Интерфейса запросов* запрос передачи осуществляется через специальные регистры PDMA.

Интерфейс прерываний отвечает за формирование сигналов прерываний при выполнении определённых событий PDMA. Например, по завершению заданной пересылки.

Следует выделить особый вид *Источника/Приёмника* - *Память*.

Память - *Источник* или *Приёмник*, который всегда готов к осуществлению передач. Для этого типа устройств не требуется *Интерфейс запросов*. *Памятью* рекомендуется считать устройство, которое при обращении к нему вставляет не более 16 циклов ожидания на шине AHB (сигнал HREADY). Если требуется более 16-ти циклов ожидания предпочтительнее использовать *Интерфейс запросов*.

47.1.1 Пересечение 1K границы

Согласно протоколу AHB непрерывная пересылка AHB не должна пересекать 1K границу адресов. PDMA автоматически разрешает эту ситуацию. В случае пересылки с пересечением 1K границы PDMA автоматически разобьёт её на две, первая из которых закончится на границе 1K, а следующая начнётся после.

47.2 Организация пересылок PDMA

Настроенный и запущенный в работу *Канал* осуществляет *PDMA Пересылку*. *PDMA Пересылка* - это наиболее крупная высокоуровневая задача по передаче данных от *Источника* к *Приёмнику*.

PDMA Пересылка состоит из *Блоков*. *Блок* - блок данных, который необходимо передать от *Источника* к *Приёмнику*. ПО управляет количеством *Блоков* в *PDMA Пересылке*. По окончанию *PDMA Пересылки Канал* завершает работу, при необходимости генерируется прерывание. После этого ПО может перенастроить *Канал* на новую *PDMA Пересылку*.

Существуют следующие виды *PDMA Пересылок*:

1. Одноблочная Пересылка.
2. Пересылка цепочкой Блоков (связным списком).
3. Пересылка с автоповтором.
4. Пересылка смежными блоками.

Источник и *Приёмник* можно независимо настроить на необходимый тип *PDMA Пересылки*.

Размер *Блока* задаётся программно, если он известен заранее (размером *Блока* управляет PDMA). Размер *Блока* задаётся количеством *Одиночных обращений* на шине AHB.

Если размер *Блока* заранее неизвестен, тогда размером *Блока* управляет периферийное устройство (*Источник* или *Приёмник*), через *Интерфейс запросов*.

Замечание: Все *транзакции AHB* (тип *транзакции AHB* определяется сигналом HBURST) состоят из *Одиночных обращений AHB* (beat AHB). Например, *транзакция* типа INCR8 состоит из 8-ми *Одиночных обращений*, а *транзакция* типа SINGLE состоит из одного *Одиночного обращения. Транзакция* типа SINGLE - *Одиночная транзакция AHB. Транзакции* других типов (INCR, INCR8 и др.) - *Пакетные транзакции AHB.*

Пересылка *Блока* от *Источника* к PDMA, а затем от PDMA к *Приёмнику* осуществляется не непрерывным потоком, а *пакетами PDMA* и *одиночными обращениями PDMA*. *Блок* разбивается на *пакеты PDMA и одиночные обращения PDMA*. *Пакет PDMA* — порция данных заданного размера, на которые разбивается *Блок PDMA*.

Размер *пакета PDMA* (другими словами, количество *одиночных обращений AHB* внутри *пакета PDMA)* настраивается программно. Если *Блок* невозможно разбить на целое количество пакетов, тогда данные, не уместившиеся в пакет, можно переслать *одиночными обращениями PDMA*. *Одиночное обращение PDMA* соответствует *одиночной транзакции AHB*. *Пакет PMDA* или *Одиночное обращение PDMA - PDMA транзакция*. *Пакет PDMA* состоит из *пакетных* и *одиночных транзакций AHB*.

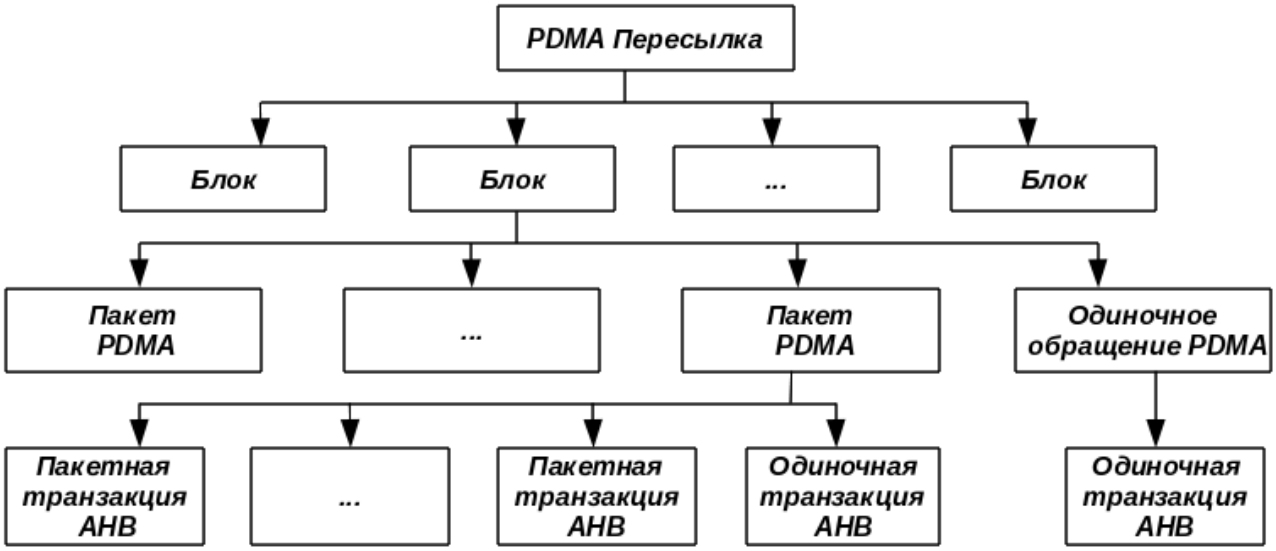


Рисунок 47.2. Организация пересылок PDMA

Таким образом, *PDMA Пересылка* состоит из одного или нескольких *Блоков*. *Блок* разбивается на *Пакеты PDMA* заданного размера. *Пакеты PDMA* преобразуются в *транзакции AHB*.

Для периферийных устройств типа *Память*, структура пересылок выглядит несколько иначе:

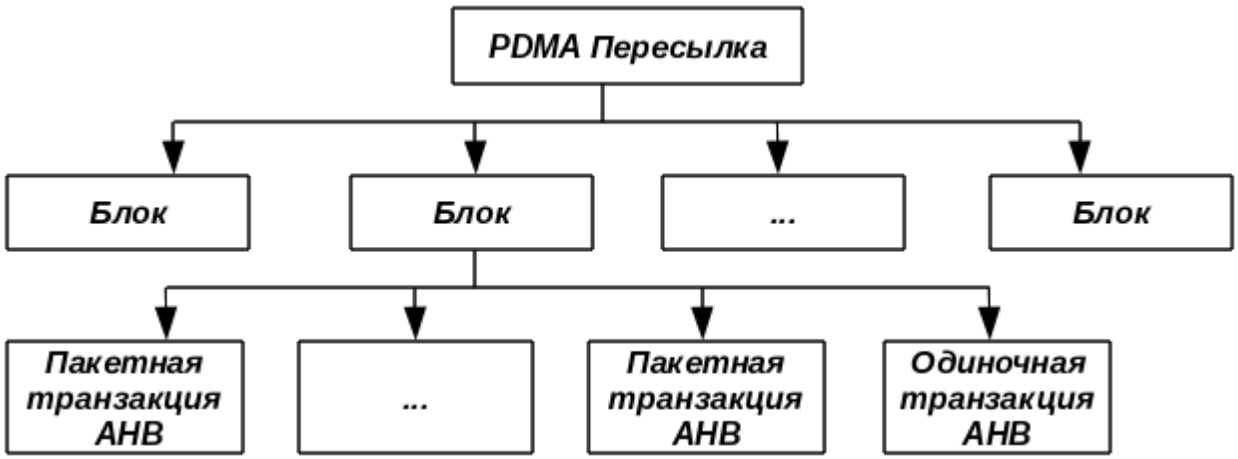


Рисунок 47.3. Организация пересылок PDMA для *Памяти*

В пересылках между PDMA и *Памятью* отсутствует уровень *Пакетов PDMA*. Блоки напрямую преобразуются в *транзакции AHB*, так как подразумевается, что *Память* всегда готова к обмену. При пересылках между *Памятью и PDMA Интерфейс запросов* не используется.

Для обычного *Источника/Приёмника* инициирование передачи *пакетов PDMA* происходит по готовности периферийного устройства (Например, по заполнению FIFO

Источника/Приёмника через *Интерфейс запросов* формируется сигнал готовности периферии к обмену).

47.3 Интерфейсы запросов

Интерфейс запросов используется при пересылке между PDMA и периферийным устройством (*Источником*/*Приёмником*) отличным от *Памяти*. Через *Интерфейс запросов* периферийное устройство сообщает PDMA о своей готовности к выдаче/приёму данных. Существуют *Программный* и *Аппаратный Интерфейсы запросов*. Для каждого *Канала* программно настраивается используемый Интерфейс запросов (как для *Источника*, так и для *Приёмника*).

47.3.1 Область одиночных обращений

В некоторых ситуациях *Блок* PDMA невозможно передать, используя только *Пакеты* PDMA. Например, размер *Блока* PDMA такой, что его невозможно передать за целое количество *Пакетов* PDMA. Тогда PDMA передаёт *Пакеты* до тех пор, пока количество оставшихся данных не станет меньше размера *Пакета*. В этом случае считается, что периферия переходит в *Область Одиночных Обращений*. Оставшиеся данные *Блока* передаются, используя *Одиночные Обращения PDMA*.

Замечание: существует возможность завершить передачи в *Области Одиночных Обращений*, используя *Прерванные Пакеты* PDMA. PDMA обрабатывает только запросы *Пакетов*, игнорируя запросы *Одиночных Обращений* PDMA, если периферия не находится в *Области Одиночных Обращений*.

Область Одиночных Обращений определяется только для периферии, которая не управляет размером *Блока*.

47.3.2 Прерванный пакет PDMA

Запросить *Пакет* PDMA возможно, даже если *Источник* или *Приёмник* находятся в *Области Одиночных Обращений*. В этом случае *Пакет* PDMA будет запущен, но будет считано только необходимое количество данных. Затем *Пакет* прервётся, не обрабатывая все запрограммированное количество байт в Пакете (поля SRC_MSIZE, DST_MSIZE, SRC_TR_WIDTH, DST_TR_WIDTH регистров CTLx).

Прерванный Пакет PDMA определяется только для периферии, которая не управляет размером *Блока*.

47.3.3 Аппаратный интерфейс запросов

Как для *Источника,* так и *Приёмника* каждого *Канала* можно назначить любой из аппаратных *Интерфейсов запросов.*

47.3.3.1 Обмен по готовности Источника/Приёмника

При готовности к обмену периферийное устройство устанавливает сигнал запроса. После получения сигнала запроса PDMA инициирует передачу на шине AHB (передачу *пакета PDMA* или о*диночного обращения PDMA*). После выполнения необходимого обмена контроллер PDMA сообщает об этом периферии. Периферийное устройство снимает сигнал запроса.

Для более эффективного использования шины AHB размер *пакетов PDMA* должен соответствовать количеству доступных ресурсов периферии (например, размеру FIFO периферии, по заполнению которого, периферия формирует сигнал готовности к обмену).

47.3.3.2 Управление размером Блока

Если размером *Блока* управляет периферийное устройство (т.е. размер *Блока* заранее не известен), то при выдаче/приёме необходимого количества данных, периферийное устройство формирует признак окончания *Блока.* После чего PDMA прекращает дальнейшие обращения к периферийному устройству, завершает обработку *Блока*, информирует об этом периферийное устройство.

Если размером *Блока* управляет PDMA (т.е. размер *Блока* известен заранее и задаётся через регистры PDMA), тогда сигнал завершения *Блока* от периферии не используется.

47.3.4 Программный Интерфейс запросов

При использовании *программного Интерфейса запросов* определение готовности периферии к обмену, а также инициирование передач производится программно. Например, периферийное устройство может сформировать сигнал прерывания, а обработчик прерывания, считав состояния периферии, программно запустить необходимые передачи.

Управление передачами осуществляется через следующие регистры:

5. ReqSrcReg - Запрос передачи *пакета PDMA Источника*;
6. SglReqSrcReg - Запрос одиночного обращения PDMA к *Источнику*;
7. LstSrcReg - Признак последней передачи (*пакета PDMA* или *одиночного обращения PDMA*) *Источника*;
8. ReqDstReg, SglReqDstReg, LstDstReg - регистры *Приёмника*, аналогичные регистрам ReqSrcReg, SglReqSrcReg, LstSrcReg *Источника*.

47.3.4.1 Программный Интерфейс запросов, периферия не управляет размером Блока

Регистры LstSrcReg и LstDstReg в этом режиме не используются и их значения игнорируются.

47.3.4.2 Источник/Приёмник находится вне Области Одиночных Обращений

Для того, чтобы инициировать пересылку *Пакета Источника* для *Канала N* необходимо установить соответствующие *Каналу* разряды запросов в регистрах ReqSrcReq и SglReqSrcReq. Т.е для запуска *Пакета* необходимо выполнение следующих условий:

ReqSrcReq[*N*] = 1 (*Запрос передачи Пакета*)

SglReqSrcReq[*N*] = 1 (З*апрос Одиночного Обращения*)

Последовательность обращений к регистрам ReqSrcReq и SglReqSrcReq вне *Области Одиночных Обращений* не важна. После выполнения передачи *Пакета*, установленные раннее разряды запросов в регистрах ReqSrcReq и SglReqSrcReq сбросятся аппаратно. Для *Приёмника* все работает аналогично (вместо регистров ReqSrcReq и SglReqSrcReq используются регистры ReqDstReq и SglReqDstReq).

47.3.4.3 Источник/Приёмник находится в Области Одиночных Обращений

Для того, чтобы инициировать пересылку *Одиночного Обращения Источника* для *Канала N* необходимо установить соответствующий *Каналу* разряд запроса в регистре SglReqSrcReq:

SglReqSrcReq[*N*] = 1 (*Запрос Одиночного Обращения*)

После этого запись запроса в ReqSrcReq[*N*] будет проигнорирована. Разряды запросов в регистрах ReqSrcReq и SglReqSrcReq будут сброшены аппаратно после завершения *Одиночного Обращения PDMA*.

Для того, чтобы запустить *Пакет* PDMA в *Области Одиночных Обращений* необходимо сначала установить разряд запроса в регистре ReqSrcReq, а затем в регистре SglReqSrcReq:

ReqSrcReq[*N*] = 1 (*Запрос передачи Пакета*)

SglReqSrcReq[*N*] = 1 (*Запрос Одиночного Обращения*)

Запущенный *Пакет* будет обработан как *Прерванный Пакет PDMA*. Разряды запросов в регистрах ReqSrcReq и SglReqSrcReq будут сброшены аппаратно после завершения *Прерванного Пакета PDMA*.

Для *Приёмника* все работает аналогично (вместо регистров ReqSrcReq и SglReqSrcReq используются регистры ReqDstReq и SglReqDstReq).

Таким образом, ПО может определять моменты завершения выполнения *Пакетов и Одиночных обращений или* опрашивая регистры ReqSrcReq/ReqDstReq, SglReqSrcReq/SglReqDstReq, или настроив прерывания IntSrcTran, IntDstTran. Прерывания генерируются по выполнению *транзакции* PDMA (как по завершению *Одиночного Обращения* PDMA, так и *Пакета* PDMA).

47.3.4.4 Программный Интерфейс запросов, периферия управляет размером Блока

Регистры ReqSrcReq/ReqDstReq, SglReqSrcReq/SglReqDstReq используются несколько иначе, чем случае, когда размером *Блока* управляет не периферия. Сначала при необходимости устанавливается LstSrcReg / LstDstReg или SglReqSrcReq / SglDstSrcReq. После этого записью в ReqSrcReq / ReqDstReq инициируется передача. Тип передачи определяется заранее предустановленными SglReqSrcReq / SglDstSrcReq (Запрос Одиночного Обращения) и LstSrcReg/ LstDstReg (Признак последней передачи в *Блоке)*. Порядок записи в регистры LstSrcReg/ LstDstReg и SglReqSrcReq/SglDstSrcReq не важен. Запись в ReqSrcReq/ ReqDstReq всегда должна следовать после записей в SglReqSrcReq/SglDstSrcReq и LstSrcReg/ LstDstReg. После завершения *транзакции PDMA* (П*акета или Одиночного обращения*) разряды запросов в регистрах *Интерфейса запроса* аппаратно сбрасываются. Если периферия управляет размером *Блока*, и блок не умещается в целое количество *Пакетов Источника* или *Приёмника*, ПО должно использовать *Одиночные Обращения*, чтобы завершить *Блок*.

47.4 Виды PDMA Пересылок

47.4.1 Одноблочная Пересылка

В случае одноблочной пересылки после выполнения передачи *Блока Канал* останавливает свою работу.

47.4.2 Пересылка цепочкой Блоков (связным списком)

Если для *Источника* или *Приёмника* настроена передача цепочкой *Блоков*, тогда после завершения передачи *Блока* следующий *Блок* автоматически проинициализируется из *Описателя Блока (LLI - linked list Item)*, адрес которого задаётся в регистре LLPx.

Описатель Блока (он же элемент связного списка или LLI) представляет из себя следующую область памяти:

Таблица 47.1. Структура Описателя Блока (LLI)

Адрес	Название	Описание
{LLPx[31:2], 2‘b00} + 0x18	LLI.DSTAT	Статус *Приёмника*. Сюда после завершения *Блока* PDMA может записать значение, считанное по адресу указанному в регистре DSTATARx.
{LLPx[31:2], 2‘b00} + 0x14	LLI.SSTAT	Статус *Источника*. Сюда после завершения *Блока* PDMA может записать значение, считанное по адресу указанному в регистре SSTATAR .
{LLPx[31:2], 2‘b00} + 0x10	LLI.CTL[63:32]	Управляющий регистр *Блока*
{LLPx[31:2], 2‘b00} + 0xC	LLI.CTL[31:0]	Управляющий регистр *Блока*
{LLPx[31:2], 2‘b00} + 0x8	LLI.LLP	Указатель на следующий *Описатель Блока*
{LLPx[31:2], 2‘b00} + 0x4	LLI.DAR	Начальный адрес *Приёмника*
{LLPx[31:2], 2‘b00}	LLI.SAR	Начальный адрес *Источника*

Из этой области инициализируются соответствующие регистры *Канала* при запуске *Блока*.

47.4.2.1 Пересылка с автоповтором

Если для *Источника* или *Приёмника* настроена *Пересылка с автоповтором*, тогда содержимое регистров *Канала*, после завершения передачи *Блока* будет обновляться значениями, которые содержались в них в момент запуска *Канала*.

47.4.2.2 Пересылка смежными Блоками

После выполнения текущего *Блока*, начало следующего *Блока* является продолжением текущего.

Возможны следующие варианты многоблочных *PDMA Пересылок*:

1. автоповтор Блока Источника - Автоповтор Блока Приёмника;
2. автоповтор Блока Источника - Цепочка Блоков Приёмника;
3. автоповтор Блока Источника - Смежные Блоки Приёмника;
4. цепочка Блоков Источника - Автоповтор Блока Приёмника;
5. цепочка Блоков Источника - Цепочка Блоков Приёмника;
6. цепочка Блоков Источника - Смежные Блоки Приёмника;
7. смежные Блоки Источника - Автоповтор Блока Приёмника;
8. смежные Блоки Источника - Цепочка Блоков Приёмника.

PDMA не поддерживает многоблочную передачу типа *Смежные Блоки Источника* - *Смежные Блоки Приёмника*. Если необходима такая функциональность, следует либо использовать *Блоки* большего размера, либо использовать пересылки *Цепочками Блоков* для *Источника* и *Приёмника* с необходимыми настройками адресов SARx и DARx.

Тип *Пересылки* определяется полями регистров LLPx, CTLx, CFGx. Содержимое регистров SARx, DARx, CTLx, LLPx *Канала* может автоматически обновляться в начале *Блока* в зависимости от типа *Пересылки*. В следующей таблице приведены возможные типы *Пересылок* и варианты изменений регистров *Канала*.

Таблица 47.2. Типы пересылок и обновление регистров *Канала*

Тип	LOC = 0 LLPx.	CTLx.LLP_SRC_EN	CFGx.RELOAD_SRC	CTLx.LLP_DST_EN	CFGx.RELOAD_DST	Обновление CTLx и LLPx	Обновление SARx	Обновление DARx	Обратная запись
Одноблочная *Пересылка* или последний *Блок* многоблочной									
1	Да	0	0	0	0	Нет	Нет	Нет	Нет
Многоблочные *Пересылки* без использования цепочек *Блоков*									
2	Да	0	0	0	1	*Нач.*	*Смеж.*	*Нач.*	Нет
3	Да	0	1	0	0	*Нач.*	*Нач.*	*Смеж.*	Нет
4	Да	0	1	0	1	*Нач.*	*Нач.*	*Нач.*	Нет
Одноблочная *Пересылка* или последний *Блок* многоблочной									
5	Нет	0	0	0	0	Нет	Нет	Нет	Да
Многоблочные *Пересылки* цепочками *Блоков*									
6	Нет	0	0	1	0	*LLI*	*Смеж.*	*LLI*	Да
7	Нет	0	1	1	0	*LLI*	*Нач.*	*LLI*	Да
8	Нет	1	0	0	0	*LLI*	*LLI*	*Смеж.*	Да
9	Нет	1	0	0	1	*LLI*	*LLI*	*Нач.*	Да
10	Нет	1	0	1	0	*LLI*	*LLI*	*LLI*	Да

В таблице используются следующие условные обозначения и сокращения:

Нач. - начальное значение. При старте *Блока* с автоповтором, в регистры автоматически записывается значение, которое было записано в них до запуска *Блока*.

Смеж. - смежные адреса *Блоков.*

LLI - значение регистра загружается и LLI (из Описателя Блока). Используется при обработке *Блока* в цепочке *Блоков.*

Обратная запись (write back) - Под обратной записью подразумевается запись управляющей и статусной информации контроллером PDMA в *Описатель Блока* при

завершении обработки *Блока*. PDMA может обновлять поля DSTAT, SSTAT, CTL LLI по завершению обработки *Блока*.

47.5 Изменение адреса внутри *Блока*

При выполнении пересылки *Блока* существует несколько возможностей изменения адреса *Источника*/*Приёмника* внутри *Блока*. Способ изменения адреса *Источника*/*Приёмника* внутри *Блока* настраивается независимо как для *Источника*, так и для *Приёмника*.

47.5.1 Адрес не меняется

Все обращения PDMA к *Источнику*/*Приёмнику* производятся по одному и тому же адресу. Например, такой способ может подойти, если в качестве буфера данных *Источника*/*Приёмника* используется FIFO.

47.5.2 Адрес увеличивается

В этом случае все обращения PDMA к *Источнику*/*Приёмнику* производятся по последовательным адресам с возрастанием адреса.

47.5.3 Адрес уменьшается

Обращения PDMA к *Источнику*/*Приёмнику* производятся по последовательным адресам с убыванием адреса.

47.5.4 Пересылка с Разбросом/Сбором

Для пересылок с увеличением или уменьшением адреса можно настроить дополнительные возможности *Сбора*(gather) и *Разброса*(scatter) *данных*.

Возможность *разброса данных* относится только к *Приёмнику*.

Для *Блока,* с включённым *Разбросом данных,* программно определяется количество *Одиночных обращений AHB* к *Приёмнику*, после которых адрес обращения увеличивается/уменьшается на заданную величину.

Сбор данных относится только к *Источнику*.

Для *Блока,* с включённым *Сбором данных,* программно определяется количество *Одиночных обращений AHB* к *Источнику*, после которых адрес обращения увеличивается/уменьшается на заданную величину.

Граница Разброса/Сбора — адрес, после которого происходит скачкообразное изменение адреса *Приёмника/Источника*. Граница достигается после заданного количества обращений со стороны PDMA(см. SGR.SGI и DSR.DSI).

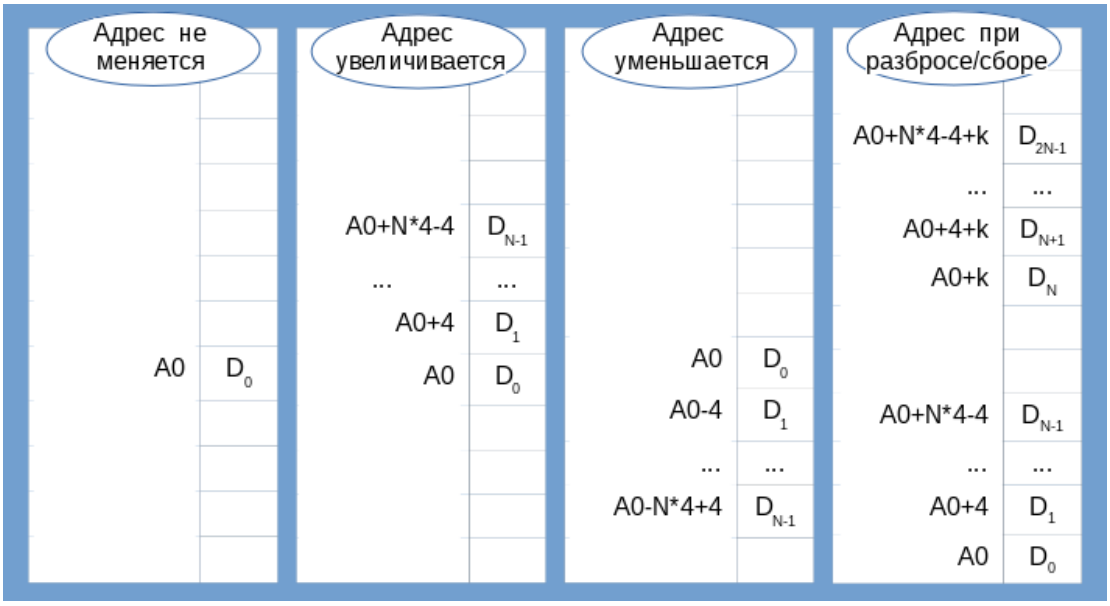


Рисунок 47.4. Пример изменения адреса в Блоке для 32-х разрядных пересылок

На Рисунок 47.5 показан пример изменения адреса *Источника* при включённой возможности *Сбора*:

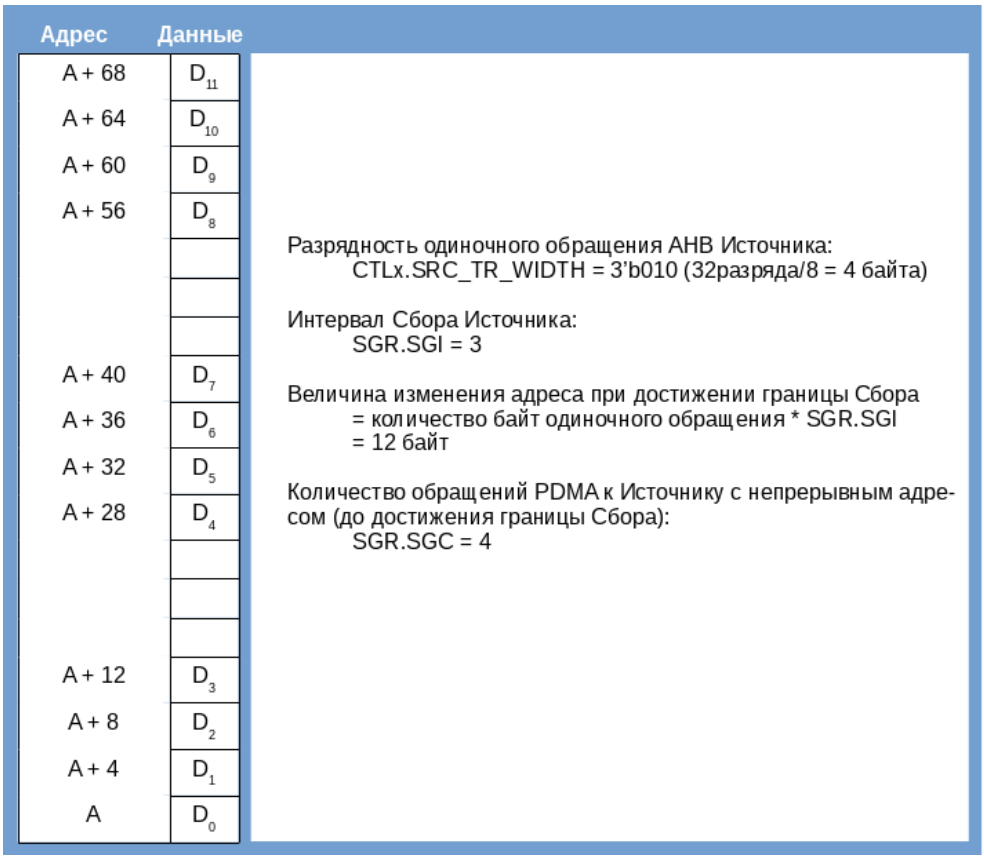


Рисунок 47.5. Пример изменения адреса для пересылки со Сбором

В примере PDMA последовательно считывает из *Источника* 4 слова (задаётся в SGR.SGC). Затем адрес обращения изменяется скачкообразно (задаётся в SGR.SGI), и следующие 4 слова считываются не из последовательного адреса, а из этого нового адреса и т.д.

При включённом *Разбросе* в *Приёмнике* обращения будут происходить аналогично.

47.6 Выключение канала до завершения пересылки

При обычном порядке работы Канал включается программно (записью единицы в ChEnReg.CH_EN) и аппаратно отключается после выполнения пересылки (ChEnReg.CH_EN сбрасывается аппаратно).

Чтобы избежать потери данных при необходимости программной остановки Канала до завершения пересылки, рекомендуется использовать следующую последовательность действий:

1. установить разряд CFGx.CH_SUSP (запрос прекращения пересылок от Источника, после этого FIFO канала не будет получать новых данных);
2. опрашивать разряд CFGx.FIFO_EMPTY до тех пор, пока FIFO не опустошится;
3. программно сбросить разряд ChEnReg.CH_EN.

Если разрядность передач от *Источника*(CTLx.SRC_TR_WIDTH) меньше разрядности передач к *Приёмнику*(CTLx.DST_TR_WIDTH) и был установлен разряд CFGx.CH_SUSP при непустом FIFO, может возникнуть ситуация, когда данные считанные из *Источника* не будут переданы *Приёмнику* (считано недостаточно разрядов для одиночного обращения к *Приёмнику*). Разрешается возобновить работу канала (записью нуля в CFGx.CH_SUSP).

Замечание: При программной остановке *Канала* не гарантируется подтверждение на запрос одиночного или пакетного обращения.

Канал также может быть остановлен без предварительных действий просто записью нуля в ChEnReg.CH_EN. Запись в CH_EN следует рассматривать как запрос на прекращение работы. Чтобы убедиться, что канал действительно остановлен, следует опрашивать ChEnReg.CH_EN до тех пор, пока из него не будет считан ноль.

Возможно, прервать работу всех каналов сразу сбросом глобального разрешения работы PDMA DmaCfgReg[0]. Для того, чтобы убедится, что каналы остановлены, необходимо опрашивать регистр ChEnReg.

Замечание: Если разряд разрешения работы сброшен при непустом FIFO, данные из FIFO *Приёмнику* не передаются. Это может привести к потере данных. При программной остановке *Канала* не гарантируется подтверждение на запрос одиночного или пакетного обращения.

47.7 Прерывания

Для каждого *Канала* существует пять источников прерываний:

1. **IntTfr** - Завершение *PDMA Пересылки.*

Прерывание формируется после завершения всей *PDMA Пересылки Приёмнику*.

2. **IntBlock** - Завершение передачи *Блока.*

Прерывание формируется после передачи всего *Блока Приёмнику*.

3. **IntDstTran** - Завершение передачи *Приёмнику.*

Формируется после выполнения последней AHB транзакции в *Пакете PDMA* или *Одиночном обращении PDMA Приёмника. Пакет* или *Одиночное обращение PDMA* инициируются через *Интерфейс запросов* (программный или аппаратный). Если *Приёмник* является устройством типа *Память*, тогда это прерывание не будет формироваться, так как для *Памяти* не существует разбиения *Блока* на *Пакеты* и *Одиночные обращения PDMA*.

4. **IntSrcTran** - Завершение передачи *Источника.*

Формируется после выполнения последней AHB транзакции в *Пакете PDMA* или *Одиночном обращении PDMA Источника. Пакет* или *Одиночное обращение PDMA* инициируются через *Интерфейс запросов* (программный или аппаратный). Если *Источник* является устройством типа *Память*, тогда это прерывание не будет формироваться, так как для *Памяти* не существует разбиения *Блока* на *Пакеты* и *Одиночные обращения PDMA*.

5. **IntError** - Ошибка передачи AHB.

Это прерывание формируется, если во время выполнения передачи через ведущий порт AHB будет получен ответ ERROR (см. сигнал HRESP AHB). Дополнительно происходит отмена *PDMA Пересылки* и отключение *Канала*.

Для работы с прерываниями используются следующие группы регистров:

1. немаскируемые регистры прерываний.

RawBlock, RawDstTran, RawErr, RawSrcTran, RawTfr. Регистры содержат информацию о выполнении условий формирования прерываний для каждого канала. Всегда устанавливаются при выполнении условия прерывания (независимо от CTLx.INT_EN).

2. регистры масок прерываний.

MaskBlock, MaskDstTran, MaskErr, MaskSrcTran, MaskTfr.

3. регистры масок прерываний каналов для всех типов прерываний;

4. маскируемые регистры прерываний.

StatusBlock, StatusDstTran, StatusErr, StatusSrcTran, StatusTfr.

Содержимое регистров Raw* маскируется регистрами Mask* и записывается в регистры Status*. Для того, чтобы прерывания могли формироваться в регистрах Status*, необходимо также глобально разрешить формирование сигналов всех прерываний через CTLx.INT_EN.

5. регистры сброса прерываний.

ClearBlock, ClearDstTran, ClearErr, ClearSrcTran, ClearTfr.

Запись в регистры Clear* очищает запросы прерываний в регистрах Raw* и Status* на одном и томже такте.

6. объединённый регистр прерываний StatusInt.

Каждый из пяти значащих разрядов регистра StatusInt содержит взятое по ИЛИ содержимое соответствующего регистра StatusBlock, StatusDstTran, StatusErr, StatusSrcTran, StatusTfr. Таким образом, регистр StatusInt указывает на наличие прерываний определённого типа. Для того, чтобы определить конкретный *Канал* прерывания, необходимо считать значения регистров StatusBlock, StatusDstTran, StatusErr, StatusSrcTran, StatusTfr.

Физические выводы сигналов прерываний:

1. IntTfr - Завершение *PDMA Пересылки* в любом *Канале;*
2. IntBlock - Завершение передачи *Блока* в любом *Канале;*
3. IntDstTran - Завершение передачи Приёмнику в любом Канале;
4. IntSrcTran - Завершение передачи Источника в любом Канале;
5. IntErr - Ошибка передачи AHB в любом Канале;
6. IntCombined - взятые по ИЛИ IntTfr, IntBlock, IntDstTran, IntSrcTran, IntErr;
7. запросы прерываний на физических выводах формируются, если прерывания разрешены через CTLx.INT_EN и установлены соответствующие маски в регистрах Mask*.

47.8 Регистры PDMA

Все регистры AHB DMA 64-х разрядные. В основном используются только младшие 32 разряда. Записи в зарезервированные разряды регистров игнорируются. При чтении в зарезервированных разрядах возвращаются нули.

47.8.1 Ошибки при доступе к регистрам

Следующие обращения к регистрам PDMA считаются запрещёнными и будут вызывать отклик на шине AHB с установленным признаком ошибки:

1. попытка обращения по AHB с hsize более 64;
2. попытка обращения по несуществующему адресу при установленному hsel;
3. запись в регистры SARx, DARx, LLPx, CTLx, SSTATx, DSTATx, SSTATARx, DSTATARx, SGRx, DSRx при включенном канале;
4. попытка чтения из регистров ClearBlock, ClearDstTran, ClearErr, ClearSrcTran, ClearTfr (доступны только по записи);
5. попытка записи в регистры StatusBlock, StatusDstTran, StatusErr, StatusSrcTran, StatusTfr, StatusInt, DmaIdReg, DMA_Component_ID_Register (доступны только по чтению).

47.8.2 Карта памяти

Таблица 47.3. Список регистров PDMA

Сокращённое название	Смещение	Тип	Название
Регистры *Канала* 0			
SAR0	0x000	RW	Адрес Источника
DAR0	0x008	RW	Адрес Приёмника
LLP0	0x010	RW	Адрес описателя следующего Блока
CTL0	0x018	RW	Регистр управления
SSTAT0	0x020	RW	Статус Источника
DSTAT0	0x028	RW	Статус Приёмника
SSTATAR0	0x030	RW	Адрес Статуса Источника
DSTATAR0	0x038	RW	Адрес Статуса Приёмника
CFG0	0x040	RW	Регистр конфигурации
SGR0	0x048	RW	Регистр Сбора Источника
DSR0	0x050	RW	Регистр Разброса Приёмника
Регистры *Канала* 1			
SAR1	0x058	RW	См. регистры *Канала* 0
DAR1	0x060	RW	См. регистры *Канала* 0
LLP1	0x068	RW	См. регистры *Канала* 0
CTL1	0x070	RW	См. регистры *Канала* 0
SSTAT1	0x078	RW	См. регистры *Канала* 0
DSTAT1	0x080	RW	См. регистры *Канала* 0
SSTATAR1	0x088	RW	См. регистры *Канала* 0
DSTATAR1	0x090	RW	См. регистры *Канала* 0
CFG1	0x098	RW	См. регистры *Канала* 0
SGR1	0x0A0	RW	См. регистры *Канала* 0
DSR1	0x0A8	RW	См. регистры *Канала* 0
Регистры *Канала* 2			
SAR2	0x0B0	RW	См. регистры *Канала* 0
DAR2	0x0B8	RW	См. регистры *Канала* 0
LLP2	0x0C0	RW	См. регистры *Канала* 0
CTL2	0x0C8	RW	См. регистры *Канала* 0
SSTAT2	0x0D0	RW	См. регистры *Канала* 0
DSTAT2	0x0D8	RW	См. регистры *Канала* 0
SSTATAR2	0x0E0	RW	См. регистры *Канала* 0
DSTATAR2	0x0E8	RW	См. регистры *Канала* 0
CFG2	0x0F0	RW	См. регистры *Канала* 0
SGR2	0x0F8	RW	См. регистры *Канала* 0
DSR2	0x100	RW	См. регистры *Канала* 0
Регистры *Канала* 3			
SAR3	0x108	RW	См. регистры *Канала* 0
DAR3	0x110	RW	См. регистры *Канала* 0

Сокращённое название	Смещение	Тип	Название
LLP3	0x118	RW	См. регистры *Канала* 0
CTL3	0x120	RW	См. регистры *Канала* 0
SSTAT3	0x128	RW	См. регистры *Канала* 0
DSTAT3	0x130	RW	См. регистры *Канала* 0
SSTATAR3	0x138	RW	См. регистры *Канала* 0
DSTATAR3	0x140	RW	См. регистры *Канала* 0
CFG3	0x148	RW	См. регистры *Канала* 0
SGR3	0x150	RW	См. регистры *Канала* 0
DSR3	0x158	RW	См. регистры *Канала* 0
Регистры *Канала* 4			
SAR4	0x160	RW	См. регистры *Канала* 0
DAR4	0x168	RW	См. регистры *Канала* 0
LLP4	0x170	RW	См. регистры *Канала* 0
CTL4	0x178	RW	См. регистры *Канала* 0
SSTAT4	0x180	RW	См. регистры *Канала* 0
DSTAT4	0x188	RW	См. регистры *Канала* 0
SSTATAR4	0x190	RW	См. регистры *Канала* 0
DSTATAR4	0x198	RW	См. регистры *Канала* 0
CFG4	0x1A0	RW	См. регистры *Канала* 0
SGR4	0x1A8	RW	См. регистры *Канала* 0
DSR4	0x1B0	RW	См. регистры *Канала* 0
Регистры *Канала* 5			
SAR5	0x1B8	RW	См. регистры *Канала* 0
DAR5	0x1C0	RW	См. регистры *Канала* 0
LLP5	0x1C8	RW	См. регистры *Канала* 0
CTL5	0x1D0	RW	См. регистры *Канала* 0
SSTAT5	0x1D8	RW	См. регистры *Канала* 0
DSTAT5	0x1E0	RW	См. регистры *Канала* 0
SSTATAR5	0x1E8	RW	См. регистры *Канала* 0
DSTATAR5	0x1F0	RW	См. регистры *Канала* 0
CFG5	0x1F8	RW	См. регистры *Канала* 0
SGR5	0x200	RW	См. регистры *Канала* 0
DSR5	0x208	RW	См. регистры *Канала* 0
Регистры *Канала* 6			
SAR6	0x210	RW	См. регистры *Канала* 0
DAR6	0x218	RW	См. регистры *Канала* 0
LLP6	0x220	RW	См. регистры *Канала* 0
CTL6	0x228	RW	См. регистры *Канала* 0
SSTAT6	0x230	RW	См. регистры *Канала* 0
DSTAT6	0x238	RW	См. регистры *Канала* 0
SSTATAR6	0x240	RW	См. регистры *Канала* 0
DSTATAR6	0x248	RW	См. регистры *Канала* 0
CFG6	0x250	RW	См. регистры *Канала* 0
SGR6	0x258	RW	См. регистры *Канала* 0
DSR6	0x260	RW	См. регистры *Канала* 0
Регистры *Канала* 7			
SAR7	0x268	RW	См. регистры *Канала* 0
DAR7	0x270	RW	См. регистры *Канала* 0
LLP7	0x278	RW	См. регистры *Канала* 0
CTL7	0x280	RW	См. регистры *Канала* 0
SSTAT7	0x288	RW	См. регистры *Канала* 0
DSTAT7	0x290	RW	См. регистры *Канала* 0
SSTATAR7	0x298	RW	См. регистры *Канала* 0
DSTATAR7	0x2A0	RW	См. регистры *Канала* 0
CFG7	0x2A8	RW	См. регистры *Канала* 0
SGR7	0x2B0	RW	См. регистры *Канала* 0
DSR7	0x2B8	RW	См. регистры *Канала* 0

Сокращённое название	Смещение	Тип	Название
Регистры прерываний			
RawTfr	0x2C0	R	Немаскируемые прерывания IntTfr
RawBlock	0x2C8	R	Немаскируемые прерывания IntBlock
RawSrcTran	0x2D0	R	Немаскируемые прерывания IntSrcTran
RawDstTrn	0x2D8	R	Немаскируемые прерывания IntDstTran
RawErr	0x2E0	R	Немаскируемые прерывания IntErr
StatusTfr	0x2E8	R	Статус прерывания IntTfr
StatusBlock	0x2F0	R	Статус прерываний IntBlock
StatusSrcTran	0x2F8	R	Статус прерываний IntSrcTran
StatusDstTran	0x300	R	Статус прерываний IntDstTran
StatusErr	0x308	R	Статус прерываний IntErr
MaskTfr	0x310	RW	Маска прерываний IntTfr
MaskBlock	0x318	RW	Маска прерываний IntBlock
MaskSrcTran	0x320	RW	Маска прерываний IntSrcTran
MaskDstTran	0x328	RW	Маска прерываний IntDstTran
MaskErr	0x330	RW	Маска прерываний IntErr
ClearTfr	0x338	W	Сброс прерываний IntTfr
ClearBlock	0x340	W	Сброс прерываний IntBlock
ClearSrcTran	0x348	W	Сброс прерываний IntSrcTran
ClearDstTran	0x350	W	Сброс прерываний IntDstTran
ClearErr	0x358	W	Сброс прерываний IntErr
StatusInt	0x360	W	Объединённый регистр прерываний
Регистры программного Интерфейса Запросов			
ReqSrcReg	0x368	RW	Запрос передачи пакета Источника
ReqDstReg	0x370	RW	Запрос передачи пакета Приёмника
SglReqSrcReg	0x378	RW	Запрос одиночного обращения Источника
SglReqDstReg	0x380	RW	Запрос одиночного обращения Приёмника
LstSrcReg	0x388	RW	Признак последнего обращения Источника
LstDstReg	0x390	RW	Признак последнего обращения Приёмника
Регистры конфигурации и управления PDMA			
DmaCfgReg	0x398	RW	Конфигурационный регистр PDMA
ChEnReg	0x3a0	RW	Регистр включения каналов PDMA
DmaIdReg	0x3a8	R	ID регистр DMA
DmaTestReg	0x3b0	RW	Тестовый регистр PDMA
Регистры аппаратных параметров PDMA			
–	0x3b8		Зарезервировано
–	0x3c0		Зарезервировано
DMA_COMP_PARAMS_6	0x3c8	R	6-ой регистр параметров PDMA
DMA_COMP_PARAMS_5	0x3d0	R	5-ый регистр параметров PDMA
DMA_COMP_PARAMS_4	0x3d8	R	4-ый регистр параметров PDMA
DMA_COMP_PARAMS_3	0x3e0	R	3-ий регистр параметров PDMA
DMA_COMP_PARAMS_2	0x3e8	R	2-ой регистр параметров PDMA
DMA_COMP_PARAMS_1	0x3f0	R	1-ый регистр параметров PDMA
Dma Component ID	0x3f8	R	ID регистр компонента DMA

47.8.3 Регистры Каналов

Каждый из восьми *Каналов* PDMA имеет следующий набор регистров:

SARx, DARx, LLPx, CTLx, SSTATx, DSTATx, SSTATARx, DSTATARx, CFGx, SGRx, DSRx (x от 0 до 7).

47.8.3.1 SARx - Адрес Источника

Регистр SARx (Source address register for channel x) -

Регистр адреса *Источника* для *Канала* x.

Смещение	SAR0 – 0x000 SAR1 – 0x058 SAR2 – 0x0b0 SAR3 – 0x108 SAR4 – 0x160 SAR5 – 0x1b8 SAR6 – 0x210 SAR7 – 0x268
Тип	RW
Состояние по сбросу	0x0
Разрядность	64

Начальное значение регистра SARx задаётся или программно до включения *Канала*, или считывается из *Описателя Блока (LLI)* перед началом обработки *Блока*. Во время обработки *PDMA Пересылки* регистр содержит текущее значение адреса, по которому PDMA производит AHB чтения из *Источника*.

Подробнее о возможностях загрузки начального значения SARx при обработке *Блока* см. в таблице «Типы пересылок и обновление регистров Канала» в Разделе «Виды PDMA Пересылок».

Таблица 47.4. Описание полей регистра SARx

Разряды	Поле	Описание
63:32	Undefined	Зарезервировано.
31:0	SAR	Текущий адрес *Источника* данных. Обновляется после каждого считывания данных контроллером PDMA из *Источника*. Поле SINC регистра CTLx определяет увеличивается ли адрес, уменьшается или остаётся неизменным после каждой передачи внутри *Блока*. Адрес должен быть выровнен согласно разрядности обращений по шине AHB для *Источника* (определяется полем CTLx.SRC_TR_WIDTH).

47.8.3.2 DARx - Адрес Приёмника

Регистр DARx (Destination address register for channel x) — Регистр адреса *Приёмника* для *Канала* x.

Смещение	DAR0 – 0x008 DAR1 – 0x060 DAR2 – 0x0b8 DAR3 – 0x110 DAR4 – 0x168 DAR5 – 0x1c0 DAR6 – 0x218 DAR7 – 0x270
Тип	RW
Состояние по сбросу	0x0
Разрядность	64

Начальное значение регистра DARx задаётся или программно до включения *Канала*, или считывается из *Описателя Блока (LLI)* перед началом обработки *Блока*. Во время обработки *PDMA Пересылки* регистр содержит текущее значение адреса, по которому PDMA производит AHB записи в *Приёмник*.

Подробнее о возможностях загрузки начального значения DARx при обработке *Блока* см. в таблице «Типы пересылок и обновление регистров Канала» в Разделе «Виды PDMA Пересылок».

Таблица 47.5. Описание полей регистра DARx

Разряды	Поле	Описание
63:32	Undefined	Зарезервировано.
31:0	DAR	Текущий адрес *Приёмника* данных. Обновляется после каждой AHB записи данных контроллером PDMA в *Приёмник*. Поле DINC регистра CTLx определяет увеличивается ли адрес, уменьшается или остаётся неизменным после каждой передачи внутри *Блока*. Адрес должен быть выровнен согласно разрядности обращений по шине AHB для *Приёмника* (определяется полем CTLx.DST_TR_WIDTH).

Автоматическое выравнивание адреса Приёмника
В определённых ситуациях при *Пересылке Смежными Блоками* после окончания передачи *Блока* адрес *Приёмника* может оказаться невыровненным для передачи следующего *Блока.* В таких ситуациях PDMA автоматически выравнивает адрес *Приёмника*, перед стартом следующего *Блока*.

Пример.

1. размер *Блока* - 9. Настраивается в CTLx.BLOCK_TS;
2. разрядность передач *Источника* - 16 (полуслово). Настраивается в CTLx.SRC_TR_WIDTH;
3. разрядность передач *Приёмника* - 32 (слово). Настраивается в CTLx.DST_TR_WIDTH;
4. для *Приёмника* настроена пересылка *Смежными Блоками*.

PDMA выполнит 4 записи по целому слову в *Приёмник*, а затем запись полуслова для того, чтобы завершить передачу *Блока*. В конце передачи *Блока* адрес *Приёмника* будет выровнен к 16-битной границе, так как последней пересылкой на AHB была пересылка полуслова. Это не совпадает с заданной разрядностью обращений *Приёмника* (32 разряда). В этом случае PDMA выровняет начальный адрес следующего *Блока Приёмника* к ближайшему 32 разрядному адресу (следующему 32 разрядному адресу, если адрес настроен на увеличение, предыдущему 32-разрядному адресу, если адрес настроен на уменьшение).

PDMA автоматически выравнивает адрес *Приёмника* при одновременном выполнении следующих условий:

1. приёмник настроен на работу со Смежными Блоками;

2. разрядность обращений AHB к Приёмнику больше разрядности обращений к Источнику (DST_TR_WIDTH > SRC_TR_WIDTH);
3. невозможно обработать Блок за целочисленное количество обращений разрядности DST_TR_WIDTH к Приёмнику ((BLOCK_TS*SRC_TR_WIDTH)/DST_TR_WIDTH != целое_число).

47.8.3.3 LLPx – Адрес Описателя следующего Блока

Регистр LLPx (Linked list pointer register for channel x)- Регистр указателя на О*писатель следующего Блока* для *Канала* x.

При работе с *Пересылкой цепочками Блоков* до того, как включить *Канал* необходимо записать в этот регистр адрес, указывающий на элемент LLI (*Описатель следующего Блока*) в памяти.

Смещение	LLP0 – 0x010 LLP1 – 0x068 LLP2 – 0x0c0 LLP3 – 0x118 LLP4 – 0x170 LLP5 – 0x1c8 LLP6 – 0x220 LLP7 – 0x278
Тип	RW
Состояние по сбросу	0x0
Разрядность	64

Таблица 47.6. Описание полей регистра LLPx

Разряды	Поле	Описание
63:32	Undefined	Зарезервировано.
31:2	LOC	Адрес *Описателя следующего Блока*. Младшие два разряда адреса не сохраняются, так как предполагается, что адрес выровнен по 32-разрядной границе. Доступ к LLI всегда 32-разрядный (HSIZE=2), выровнен по 32-разрядной границе.
1:0	LMS	List Master Select. Выбор AHB интерфейса, по которому PDMA произведёт чтение LLI. 00 — AHB Master 1 01 — AHB Master 2 10 — AHB Master 3 11 — AHB Master 4 PDMA может служить мостом между несколькими различными шинами AHB (до 4-х) и осуществлять передачи между устройствами на этих разных шинах.

Регистр LLPx выполняет две функции:

1. *Определение типа Пересылки*. Нулевое значение поля LOC выключает использование *Пересылок цепочками Блоков* и позволяет задать одноблочную *Пересылку*. См. таблицу «Типы пересылок и обновление регистров *Канала*» в Разделе «Виды PDMA *Пересылок*».

2. *Хранение адреса Описателя следующего Блока или адреса Статуса Источника/Приёмника*. Если поле LOC отлично от нуля, тогда в нем содержится адрес *Описателя следующего Блока.* Регистр также может содержать адрес по которому производилась запись статуса *Источника/Приёмника* после завершения обработки Блока. См. таблицу «Структура *Описателя Блока* (LLI)» в Разделе «*Виды* PDMA *Пересылок*».

47.8.3.4 CTLx – Регистр управления

Регистр CTLx (Control register for channel x) – Регистр управления *Каналом* x.

Управление *Пересылкой PDMA* осуществляется через поля этого регистра. При использовании *Пересылок цепочками Блоков* регистр CTLx обновляется из соответствующей области *Описателя Блока* (LLI). Если включена возможность *обратной записи*, в конце передачи каждого *Блока* разряды CTLx[63:32] записываются в соответствующее местоположение *Описателя Блока*. См. таблицу «Типы пересылок и обновление регистров *Канала*» в Разделе «Виды PDMA *Пересылок*». Значение регистра CTLx необходимо задать до включения *Канала*.

Смещение	CTL0 – 0x018 CTL1 – 0x070 CTL2 – 0x0c8 CTL3 – 0x120 CTL4 – 0x178 CTL5 – 0x1d0 CTL6 – 0x228 CTL7 – 0x280
Тип	RW
Состояние по сбросу	0x00000002_00104825
Разрядность	64

Таблица 47.7. Описание полей регистра CTLx

Разряды	Поле	Описание
63:45	Undefined	Зарезервировано.
44	DONE	При использовании возможности *обратной записи* разряды CTLx[63:32] записываются в LLI в конце обработки *Блока* с установленным разрядом DONE. Программно опрашивая разряд DONE LLI.CTL, можно определить выполнен ли конкретный *Блок* в цепочке или нет. При настройке цепочки *Блоков* разряд DONE регистра LLI.CTL должен быть сброшен до запуска *Канала*.
43:32	BLOCK_TS	Размер *Блока* (Block Transfer Size). Если размером *Блока* управляет PDMA, то это поле устанавливается пользователем до запуска *Канала*. Значение поля задаёт количество *Одиночных обращений* на шине AHB к *Источнику*, которое будет произведено за один *Блок*. *Одиночное обращение* соответствует одному «beat» AHB. Разрядность *Одиночного обращения* определяется полем CTLx.SRC_TR_WIDTH. После запуска передачи BLOCK_TS возвращает текущее количество элементов данных, считанных из *Источника (*независимо от того управляет ли размером *Блока* PDMA или периферийное устройство). Если размером *Блока* управляет *Источник/Приёмник*, тогда максимальное значение, прочитанное из BLOCK_TS, ограничено параметром DMAH_CHx_MAX_BLK_SIZE. Хотя фактических чтений из *Источника* может производиться больше (размер *Блока* может быть больше). Значение по сбросу — 0x2.

<table>
<tr><th>Разряды</th><th>Поле</th><th>Описание</th></tr>
<tr><td></td><td></td><td>Замечание: Для каждого Канала аппаратно задаётся максимальный размер Блока — DMAH_CHx_MAX_BLK_SIZE. Если размером Блока управляет PDMA, тогда DMAH_CHx_MAX_BLK_SIZE определяет максимальное разрешённое значение для BLOCK_TS.</td></tr>
<tr><td>31:29</td><td>Undefined</td><td>Зарезервировано.</td></tr>
<tr><td>28</td><td>LLP_SRC_EN</td><td>Разрешение цепочки Блоков Источника. Цепочки Блоков Источника разрешены при LLPx.LOC != 0 и LLP_SRC_EN в 1.</td></tr>
<tr><td>27</td><td>LLP_DST_EN</td><td>Разрешение цепочки Блоков Приёмника. Цепочки Блоков Приёмника разрешены при LLPx.LOC != 0 и LLP_DST_EN в 1.</td></tr>
<tr><td>26:25</td><td>SMS</td><td>Source Master Select. Выбор AHB интерфейса, по которому PDMA обращается к Источнику.
00 — AHB Master 1
01 — AHB Master 2
10 — AHB Master 3
11 — AHB Master 4
PDMA может служить мостом между несколькими различными шинами AHB (до 4-х) и осуществлять передачи между устройствами на этих разных шинах.</td></tr>
<tr><td>24:23</td><td>DMS</td><td>Destination Master Select. Выбор AHB интерфейса, по которому PDMA обращается к Приёмнику.
00 — AHB Master 1
01 — AHB Master 2
10 — AHB Master 3
11 — AHB Master 4
PDMA может служить мостом между несколькими различными шинами AHB (до 4-х) и осуществлять передачи между устройствами на этих разных шинах.</td></tr>
<tr><td>22:20</td><td>TT_FC</td><td>Тип передачи и Управление размером Блока (Transfer Type and Flow Control). Возможны следующие варианты:
<table>
<tr><th>TT_FC</th><th>Тип передачи</th><th>Управление размером Блока</th></tr>
<tr><td>000</td><td>Память-Память</td><td>PDMA</td></tr>
<tr><td>001</td><td>Память-Периферия</td><td>PDMA</td></tr>
<tr><td>010</td><td>Периферия-Память</td><td>PDMA</td></tr>
<tr><td>011</td><td>Периферия-Периферия</td><td>PDMA</td></tr>
<tr><td>100</td><td>Периферия-П
мять</td><td>Периферия</td></tr>
<tr><td>101</td><td>Периферия-Периферия</td><td>Источник</td></tr>
<tr><td>110</td><td>Память-Периферия</td><td>Периферия</td></tr>
<tr><td>111</td><td>Периферия-Периферия</td><td>Приёмник</td></tr>
</table>
При использовании цепочки Блоков поле TT_FC должно оставаться неизменным для всех Блоков цепочки.

Примечание. Для каждого Канала аппаратно задан тип управления размером Блока:
Размером Блока управляет только PDMA (DMA_FC_ONLY)
Размером Блока управляет только Источник (SRC_FC_ONLY)
Размером Блока управляет только Приёмник (DST_FC_ONLY)
Размером Блока управляет или Периферия или PDMA (ANY_FC)
Значение по сбросу:
TT_FC[0] = 1
TT_FC[1] = 1 если тип отличен от SRC_FC_ONLY
TT_FC[2] = 1 при SRC_FC_ONLY или DST_FC_ONLY</td></tr>
</table>

<table>
<tr><th>Разряды</th><th>Поле</th><th>Описание</th></tr>
<tr><td></td><td></td><td>Поле TT_FC[2] не реализовано и при чтении возвращает 0 при DMA_FC_ONLY.
Поле TT_FC[2:1] не реализовано и при чтении возвращает 10 при SRC_FC_ONLY.
Поле TT_FC[2:1] не реализовано и при чтении возвращает 11 при DST_FC_ONLY.</td></tr>
<tr><td>19</td><td>Undefined</td><td>Зарезервировано.</td></tr>
<tr><td>18</td><td>DST_SCATTER_EN</td><td>Разрешение режима Разброса данных Приёмника.
0 - Разброс запрещён.
1 - Разброс разрешён.
Режим Разброса применим только в том случае, если поле CTLx.DINC настроено на увеличение или уменьшение адреса Приёмника.

Примечание. Поле не реализовано и при чтении возвращает ноль, если не задан аппаратный параметр DMAH_CHx_DST_SCA_EN.</td></tr>
<tr><td>17</td><td>SRC_GATHER_EN</td><td>Разрешение режима Сбора данных Источника.
0 - Сбор запрещён.
1 - Сбор разрешён.
Режим Сбора применим только в том случае, если поле CTLx.SINC настроено на увеличение или уменьшение адреса Источника.

Примечание. Поле не реализовано и при чтении возвращает ноль, если не задан аппаратный параметр DMAH_CHx_SRC_GAT_EN.</td></tr>
<tr><td>16:14</td><td>SRC_MSIZE</td><td>Размер пакета Источника (Source Burst Transaction Length). Определяет количество элементов данных разрядностью CTLx.SRC_TR_WIDTH, которое PDMA считывает из Источника после поступления сигнала запроса передачи пакета от программного или аппаратного Интерфейса Запросов.

Значение по сбросу - 1.

Замечание: это поле не имеет отношения к сигналам HBURST AHB.
Зависимость значения SRC_MSIZE и числа элементов в пакете:
<table>
<tr><th>SRC_MSIZE</th><th>Число элементов данных разрядностью CTLx.SRC_TR_WIDTH в пакете</th></tr>
<tr><td>000</td><td>1</td></tr>
<tr><td>001</td><td>4</td></tr>
<tr><td>010</td><td>8</td></tr>
<tr><td>011</td><td>16</td></tr>
<tr><td>100</td><td>32</td></tr>
<tr><td>101</td><td>64</td></tr>
<tr><td>110</td><td>128</td></tr>
<tr><td>111</td><td>256</td></tr>
</table>
Примечание. Аппаратный параметр DMAH_CHx_MAX_MULT_SIZE определяет максимальное значение SRC_MSIZE, которое можно задать программно и соответственно количество используемых разрядов MSIZE. При чтении неиспользуемых разрядов возвращается ноль.</td></tr>
<tr><td>13:11</td><td>DEST_MSIZE</td><td>Размер пакета Приёмника (Destination Burst Transaction Length). Определяет количество элементов данных разрядностью CTLx.DST_TR_WIDTH, которое PDMA записывает в Приёмник после поступления сигнала запроса передачи пакета от программного или аппаратного Интерфейса Запросов.</td></tr>
</table>

<table>
<tr><th>Разряды</th><th>Поле</th><th>Описание</th></tr>
<tr><td></td><td></td><td>Значение по сбросу, размер пакета и ограничения те же, что и у SRC_MSIZE.</td></tr>
<tr><td>10:9</td><td>SINC</td><td>Тип изменения адреса Источника (Source Address Increment).

Поле SINC и изменение адреса Источника после каждого обращения PDMA к Источнику:
00 - Инкремент
01 - Декремент
1x - Не меняется (подходит для работы с FIFO)

Увеличение/уменьшение адреса выровнено согласно CTLx.SRC_TR_WIDTH.</td></tr>
<tr><td>8:7</td><td>DINC</td><td>Тип изменения адреса Приёмника (Destination Address Increment).

Поле DINC и изменение адреса Приёмника после каждого обращения PDMA к Приёмнику:
00 - Инкремент
01 - Декремент
1x - Не меняется (подходит для работы с FIFO)

Увеличение/уменьшение адреса выровнено согласно CTLx.DST_TR_WIDTH.</td></tr>
<tr><td>6:4</td><td>SRC_TR_WIDTH</td><td>Разрядность одиночного обращения AHB Источника(Source Transfer Width).
Разрядность обращения в зависимости от значения SRT_TR_WIDTH (см. HSIZE AHB):
<table>
<tr><th>SRC_TR_WIDTH</th><th>Разрядность обращения AHB (в битах)</th></tr>
<tr><td>000</td><td>8</td></tr>
<tr><td>001</td><td>16</td></tr>
<tr><td>010</td><td>32</td></tr>
<tr><td>011</td><td>64</td></tr>
<tr><td>100</td><td>128</td></tr>
<tr><td>101</td><td>256</td></tr>
<tr><td>11x</td><td>256</td></tr>
</table>
Разрядность настраиваемая через SRC_TR_WIDTH не должна превышать разрядности шины AHB мастер порта. Поле SRC_TR_WIDTH существует, если не задан аппаратный параметр DMAH_CHx_STW, жёстко определяющий разрядность обращений. Если параметр DMAH_CHx_STW задан, то при чтении поля будет возвращаться аппаратно заданное значение DMAH_CHx_STW.</td></tr>
<tr><td>3:1</td><td>DST_TR_WIDTH</td><td>Разрядность одиночного обращения AHB Приёмника (Destination Transfer Width).
Возможные значения аналогичны полю SRC_TR_WIDTH.</td></tr>
<tr><td>0</td><td>INT_EN</td><td>Разрешение прерываний (Interrupt Enable Bit).
Если установлено в 1, то включаются все возможные источники прерываний. Работает как глобальная маска для всех прерываний Канала. Но прерывания в немаскируемых(Raw*) регистрах будут формироваться, даже если поле установлено в 0.
Значение по сбросу - 1.</td></tr>
</table>

47.8.3.5 SSTATx – Статус Источника

Регистр SSTATx (Source status register for channel x) - Регистра статуса *Источника* для *Канала* x.

После завершения передачи *Блока* PDMA может прочитать информацию о состоянии *Источника* по адресу заданному в SSTATARx. Эта информация сохраняется в регистре SSTATx и записывается в местоположение SSTATx в блоке LLI до старта следующего *Блока*. Регистр является временным буфером статуса *Источника* при его записи в LLI. Программному обеспечению следует брать информацию о статусе из блока LLI, а не из этого регистра.

Смещение	SSTAT0 – 0x020 SSTAT1 – 0x078 SSTAT2 – 0x0d0 SSTAT3 – 0x128 SSTAT4 – 0x180 SSTAT5 – 0x1d8 SSTAT6 – 0x230 SSTAT7 – 0x288
Тип	RW
Состояние по сбросу	0x0
Разрядность	64

Таблица 47.8. Описание полей регистра SSTATx

Разряды	Поле	Описание
63:32	Undefined	Зарезервировано.
31:0	SSTAT	Информация о состоянии *Источника*, аппаратно загружаемая из местоположения указанного в регистре SSTATARx. Разрешение считывания статуса источника задаётся в регистре CFGx.SS_UPD_EN. Поле реализовано, если задан аппаратный параметр DMAH_CHx_STAT_SRC. Если параметр DMAH_CHx_STAT_SRC не задан, при чтении возвращается ноль.

47.8.3.6 DSTATx – Статус Приёмника

Регистр DSTATx (Destination status register for channel x) - Регистра статуса *Приёмника* для *Канала* x.

После завершения передачи *Блока* PDMA может прочитать информацию о состоянии *Приёмника* по адресу заданному в DSTATARx. Эта информация сохраняется в регистре DSTATx и записывается в местоположение DSTATx в блоке LLI до старта следующего *Блока*. Регистр является временным буфером статуса *Приёмника* при его записи в LLI. Программному обеспечению следует брать информацию о статусе из блока LLI, а не из этого регистра.

Смещение	DSTAT0 – 0x028 DSTAT1 – 0x080 DSTAT2 – 0x0d8 DSTAT3 – 0x130 DSTAT4 – 0x188 DSTAT5 – 0x1e0 DSTAT6 – 0x238 DSTAT7 – 0x290
Тип	RW
Состояние по сбросу	0x0
Разрядность	64

Таблица 47.9. Описание полей регистра DSTATx

Разряд	Имя	Описание
63:32	Undefined	Зарезервировано.
31:0	DSTAT	Информация о состоянии *Приёмника*, аппаратно загружаемая из местоположения указанного в регистре DSTATARx. Разрешение считывания статуса источника задаётся в регистре CFGx.DS_UPD_EN. Поле реализовано, если задан аппаратный параметр DMAH_CHx_STAT_DST. Если параметр DMAH_CHx_STAT_DST не задан, при чтении возвращается ноль.

47.8.3.7 SSTATARx – Адрес Статуса Источника

Регистр SSTATARx (Source status address register for channel x) - Адрес статуса *Источника* для *Канала* x.

После завершения передачи *Блока* PDMA может аппаратно считать информацию о состоянии *Источника*. Адрес статусной информации определяется пользователем в регистре SSTATARx. Пользователь может выбрать любое местоположение системной памяти, где находится 32-разрядная информация о состоянии *Источника*. Например, если в качестве периферии используется блок DW_apb_ssi, пользователь может использовать один из регистров DW_apb_ssi для индикации состояния приёмника. В SSTATARx можно задать адреса таких регистров DW_apb_ssi как: SSI.CTRL, SSI.ISR, SSI.RXFLR и т.д.

Смещение	SSTATAR0 – 0x030 SSTATAR1 – 0x088 SSTATAR2 – 0x0e0 SSTATAR3 – 0x138 SSTATAR4 – 0x190 SSTATAR5 – 0x1e8 SSTATAR6 – 0x240 SSTATAR7 – 0x298
Тип	RW
Состояние по сбросу	0x0
Разрядность	64

Таблица 47.10. Описание полей регистра SSTATARx

Разряды	Поле	Описание
63:32	Undefined	Зарезервировано.

Разряды	Поле	Описание
31:0	SSTATAR	Указывает на местоположение памяти, из которого PDMA считывает информацию о состоянии *Источника* в регистр SSTATx, а затем сохраняет её в соответствующую ячейку LLI до запуска следующего *Блока*.

47.8.3.8 DSTATARx – Адрес Статуса Приёмника

Регистр DSTATARx (Destination status address register for channel x) – адрес статуса Приёмника для Канала x.

После завершения передачи Блока PDMA может считать информацию о состоянии Приёмника. Адрес статусной информации определяется пользователем в регистре DSTATARx. Пользователь может выбрать любое местоположение системной памяти, где находится 32-разрядная информация о состоянии Приёмника. Например, если в качестве периферии используется блок DW_apb_ssi, пользователь может использовать один из регистров DW_apb_ssi для индикации состояния приёмника. В DSTATARx можно задать адреса таких регистров DW_apb_ssi как: SSI.CTRL, SSI.ISR, SSI.TXFLR и т.д.

Смещение	DSTATAR0 – 0x038 DSTATAR1 – 0x090 DSTATAR2 – 0x0e8 DSTATAR3 – 0x140 DSTATAR4 – 0x198 DSTATAR5 – 0x1f0 DSTATAR6 – 0x248 DSTATAR7 – 0x2a0
Тип	RW
Состояние по сбросу	0x0
Разрядность	64

Таблица 47.11. Описание полей регистра DSTATARx

Разряды	Поле	Описание
63:32	Undefined	Зарезервировано.
31:0	DSTATAR	Указывает на местоположение памяти, из которого PDMA считывает информацию о состоянии *Приёмника* в регистр DSTATx, а затем сохраняет её в соответствующую ячейку LLI до запуска следующего *Блока*.

47.8.3.9 CFGx – Регистр Конфигурации

Регистр CFGx (Configuration register for channel x) - Регистр конфигурации *Канала* x.

Настройка *Пересылки* осуществляется через этот регистр. Регистр конфигурации останется неизменным для всех *Блоков* многоблочной *Пересылки.* Регистр необходимо проинициализировать до включения *Канала*.

Смещение	CFG0 – 0x040 CFG1 – 0x098 CFG2 – 0x0f0 CFG3 – 0x148 CFG4 – 0x1a0 CFG5 – 0x1f8 CFG6 – 0x250 CFG7 – 0x2a8
Тип	RW
Состояние по сбросу	0x0000_0004_0000_0Ex0
Разрядность	64

Таблица 47.12. Описание полей регистра CFGx

Разряды	Поле	Описание
63:47	Undefined	Зарезервировано.
46:43	DEST_PER	Определяет аппаратный Интерфейс запросов Приёмника для Канала x, если выбрана работа с аппаратным Интерфейсом запросов (CFGx.HS_SEL_DST = 0). Поле игнорируется, если выбран программный Интерфейс запросов (CFGx.HS_SEL_DST = 1). PDMA будет взаимодействовать с Приёмником через заданный Интерфейс запросов.
42:39	SRC_PER	Определяет аппаратный интерфейс Запросов Источника для Канала x, если выбрана работа с аппаратным Интерфейсом запросов (CFGx.HS_SEL_SRC = 0). Поле игнорируется, если выбран программный Интерфейс запросов (CFGx.HS_SEL_SRC = 1). PDMA будет взаимодействовать с Источником через заданный Интерфейс запросов.
38	SS_UPD_EN	Разрешение считывания статуса Источника (Source Status Update Enable). Если SS_UPD_EN в 1, тогда информация о состоянии Источника считывается из местоположения, на которое указывает регистр SSTATARx, сохраняется в регистр SSTATx и записывается в соответствующую ячейку блока LLI.
37	DS_UPD_EN	Разрешение считывания статуса Приёмника (Destination Status Update Enable). Если DS_UPD_EN в 1, тогда информация о состоянии Приёмника считывается из местоположения, на которое указывает регистр DSTATARx, сохраняется в регистр DSTATx и записывается в соответствующую ячейку блока LLI.
36:34	PROTCTL	(Protection Control). Используется для задания сигналов HPROT[3:1] шины AHB. Спецификация AMBA рекомендует по умолчанию устанавливать «non-cached, non-buffered, privileged data access». HPROT[0] всегда в единице. По сбросу устанавливается в 0x1, что соответствует рекомендациям AMBA. Зависимость сигналов AHB от значения поля PROTCTL: HPROT[0] - 1 HPROT[1] - PROTCTL[0] HPROT[2] - PROTCTL[1] HPROT[3] - PROTCTL[2]
33	FIFO_MODE	Выбор режима FIFO(FIFO Mode Select). Определяет количество данных или свободное пространство FIFO, достаточное для обслуживания запроса передачи пакета. 0 - Для обслуживания запроса передачи пакета в FIFO должно быть достаточно данных/места для одиночной AHB передачи заданной разрядность. 1 - Для обслуживания запроса передачи пакета Приёмника FIFO должно быть заполнено на половину и более. Для обслуживания запроса приёма пакета Источника FIFO должно быть свободно более чем на половину. Исключениями являются окончание пакета или окончание блока.
32	FCMODE	Flow Control Mode. Определяет момент обслуживания запроса передачи от источника, когда размером блока управляет приёмник.

Разряды	Поле	Описание
		0 - Запросы от источника обслуживаются по мере их поступления. DMA считывает данные из источника даже если не было запросов от приёмника (т.е. предварительная выборка/data pre-fetching разрешена). 1 - Запрос от источника не обработается до тех пор, пока приёмник не установит свой запрос. В этом режиме гарантируется, что количество данных записанных в приёмник совпадёт с количеством, считанных из источника (приёмник определяет величину блока, и может прервать блок в любой момент.) Предварительная выборка данных запрещена.
31	RELOAD_DST	Автоматическое обновление адреса приёмника (Automatic Destination Reload). Поле задаёт режим автоповтора Блока для Приёмника. При пересылке с автоповтором в конце каждого блока регистр DARx автоматически возвращаться к своему начальному значению. После этого запускается передача нового блока.
30	RELOAD_SRC	Автоматическое обновление адреса источника (Automatic Source Reload). Поле задаёт режим автоповтора Блока для Источника. При пересылке с автоповтором в конце каждого блока регистр SARx автоматически возвращаться к своему начальному значению. После этого запускается передача нового блока.
29:20	MAX_ABRST	Максимальный размер пакетной транзакции AHB (Maximum AMBA Burst Length). 0 - ПО не ограничивает максимальный размер AHB транзакции. Если MAX_ABRST больше нуля, тогда количество обращений внутри пакетной транзакции AHB ограничено значением MAX_ABRST. *Замечание*: Это поле не реализовано и при чтении возвращает ноль, если не установлен аппаратный параметр DMAH_MABRST.
19	SRC_HS_POL	Активный уровень интерфейса запросов источника (Source Handshaking Interface Polarity). 0 - Активный уровень высокий. 1 - Активный уровень низкий.
18	DST_HS_POL	Активный уровень интерфейса запросов приёмника (Destination Handshaking Interface Polarity). 0 - Активный уровень высокий. 1 - Активный уровень низкий.
17	LOCK_B	Блокировка шины (Bus Lock Bit). Сигнал AHB HLOCK формируется на период, определённый в CFGx.LOCK_B_L, если установлен разряд LOCK_B. *Замечание*: Это поле не реализовано и при чтении возвращает ноль, если не установлен аппаратный параметр DMAH_CHx_LOCK_EN.
16	LOCK_CH	Блокировка канала (Channel Lock Bit). Если каналу предоставлено управление мастер интерфейсом AHB и установлен разряд CFGx.LOCK_CH, то ни один другой другой канал не получит доступ к AHB в течении периода, задаваемого в CFGx.LOCK_CH_L. *Замечание*: Это поле не реализовано и при чтении возвращает ноль, если не установлен аппаратный параметр DMAH_CHx_LOCK_EN.
15:14	LOCK_B_L	Bus Lock Level. Определяет период в течении которого применяется разряд CFGx.LOCK_B. 00 - До завершения PDMA Пересылки. 01 - До завершения передачи Блока. 1x - До завершения одиночного обращения PDMA или пакета PDMA. *Замечание*: Это поле не реализовано и при чтении возвращает ноль, если не установлен аппаратный параметр DMAH_CHx_LOCK_EN.
13:12	LOCK_CH_L	Channel Lock Level. Определяет интервал в течении которого применяется разряд CFGx.LOCK_CH. 00 - До завершения PDMA Пересылки. 01 - До завершения передачи Блока. 1x - До завершения PDMA транзакции. *Замечание*: Это поле не реализовано и при чтении возвращает ноль, если не установлен аппаратный параметр DMAH_CHx_LOCK_EN.

Разряды	Поле	Описание
11	HS_SEL_SRC	Выбор интерфейса запросов источника (Source Software or Hardware Handshaking Select). 0 - Аппаратный интерфейс. Программно инициируемые транзакции DMA игнорируются. 1 - Программный интерфейс запросов. Аппаратно инициируемые транзакции игнорируются. Если источник является памятью, тогда разряд игнорируется. Значение по сбросу - 0x1 (программный интерфейс).
10	HS_SEL_DST	Выбор интерфейса запросов приёмника (Destination Software or Hardware Handshaking Select). 0 - Аппаратный интерфейс. Программно инициируемые транзакции PDMA игнорируются. 1 - Программный интерфейс запросов. Аппаратно инициируемые транзакции игнорируются. Если приёмник является памятью, тогда разряд игнорируется. Значение по сбросу - 0x1 (программный интерфейс).
9	FIFO_EMPTY	Признак наличия данных в FIFO. Может использоваться совместно с CFGx.CH_SUSP, для «чистого» отключения канала (преждевременного, до штатного отключения). 1 - FIFO свободно. 0 - В FIFO есть слова. Значение по сбросу - 0x1.
8	CH_SUSP	Channel Suspend. Запись 1 приостанавливает все передачи данных от источника, до тех пор пока не будет записан 0. Нет гарантии, что текущая транзакция будет завершена. Можно использовать совместно с FIFO_EMPTY, чтобы остановить канал без потери данных. 0 - Нормальная работа. 1 - Приостановка передач от источника.
7:5	CH_PRIOR	Приоритет Канала (Channel priority). Приоритет 7 - наивысший. 0 - низший. Значение по сбросу - соответствует номеру канала (для 0 - 0, …, для 7 - 7).
4:0	Undefined	Зарезервировано.

47.8.3.10 SGRx – Регистр Сбора Источника

Регистр SGRx (Source gather register for channel x) - Регистр Сбора для канала x.

Если включён режим Сбора(Gather), то через определённое количество чтений из Источника по AHB, адрес чтения автоматически увеличивается/уменьшается на заданную величину. Регистр позволяет задать количество чтений, при которых адрес меняется неразрывно (последовательно увеличивается/уменьшается), а также величину, на которую изменяется адрес чтения при приёме заданного количества слов. Увеличение/уменьшение адреса чтения задаётся полем SINC регистра CTLx. Если поле SINC регистра CTLx настроено на неизменный адрес, тогда адрес не изменяется и регистр SGRx игнорируется.

Смещение	SGR0 – 0x048 SGR1 – 0x0a0 SGR2 – 0x0f8 SGR3 – 0x150 SGR4 – 0x1a8 SGR5 – 0x200 SGR6 – 0x258 SGR7 – 0x2b0
Тип	RW
Состояние по сбросу	0x0
Разрядность	64

Таблица 47.13. Описание полей регистра SGRx

Разряды	Поле	Описание
63:32	Undefined	Зарезервировано.
31:20	SGC	Счётчик Сбора Источника (Source gather count). Задаёт количество чтений AHB разрядностью CTLx.SRC_TR_WIDTH, после которых адрес чтения изменяется скачком (см. поле SGI).
19:0	SGI	Интервал Сбора Источника (Source gather interval). Определяет величину изменения адреса чтения источника после достижения границы сбора. Т.е. каждые N чтений (задаётся полем SGC) из источника адрес чтения изменяется на величину, заданную этим полем, умноженную на разрядность обращения по AHB к Источнику в байтах (см. CTLx.SRC_TR_WIDTH).

47.8.3.11 DSRx – Регистр Разброса Приёмника

Регистр DSRx (Destination scatter register for channel x) - Регистр Разброса Приёмника для канала x.

Если включён режим Разброса(Scatter), то через определённое количество записей по AHB в приёмник, адрес записи автоматически увеличивается/уменьшается на заданную величину. Регистр позволяет задать количество передач, при которых адрес меняется неразрывно (последовательно увеличивается/уменьшается), а также величину, на которую изменяется адрес передачи при выдаче заданного количества слов. Увеличение/уменьшение адреса записи задаётся полем DINC регистра CTLx. Если поле DINC регистра CTLx настроено на неизменный адрес, тогда адрес не изменяется и регистр DSRx игнорируется

Смещение	DSR0 – 0x050 DSR1 – 0x0a8 DSR2 – 0x100 DSR3 – 0x158 DSR4 – 0x1B0 DSR5 – 0x208 DSR6 – 0x260 DSR7 – 0x2b8
Тип	RW
Состояние по сбросу	0x0
Разрядность	64

Таблица 47.14. Описание полей регистра DSRx

Разряды	Поле	Описание
63:32	Undefined	Зарезервировано.
31:20	DSC	Счётчик Разброса Приёмника (Destination scatter count). Задаёт количество записей по AHB в Приёмник разрядностью CTLx.DST_TR_WIDTH, после которых адрес передачи изменяется скачком (см. поле DSI).
19:0	DSI	Интервал Разброса Приёмника (Destination scatter interval). Определяет величину изменения адреса записи приёмника при достижении границы разброса. Т.е. каждые N записей (задаётся полем DSC) в приёмник адрес передачи изменяется на величину, заданную этим полем, умноженную на разрядность обращения к Приёмнику (см. CTLx.DST_TR_WIDTH).

47.8.4 Регистры Прерываний

47.8.4.1 Регистры RawTfr, RawBlock, RawSrcTran, RawDstTran, RawErr

Перед маскированием события прерываний формируются в регистрах RawBlock, RawDstTran, RawErr, RawSrcTran, RawTfr. Каждый из регистров содержит информацию о прерываниях определённого типа

(IntTfr, IntBlock, IntSrcTran, IntDstTran, IntErr). См. Раздел «1.7 Прерывания». Каждый из разрядов этих регистров сбрасывается записью единицы в соответствующие разряды регистров ClearTfr, ClearBlock, ClearSrcTran, ClearDstTran, ClearErr. Регистры Raw* доступны по записи только для целей тестирования. В обычном режиме не рекомендуется производить запись в эти регистры.

Смещение	RawTfr - 0x2c0 RawBlock - 0x2c8 RawSrcTran - 0x2d0 RawDstTran - 0x2d8 RawErr - 0x2e0
Тип	RW
Состояние по сбросу	0x0
Разрядность	64

Таблица 47.15. Описание полей регистров RawTfr, RawBlock, RawSrcTran, RawDstTran, RawErr

Разряды	Поле	Описание
63:8	Undefined	Зарезервировано.
7:0	RAW	Разряды с седьмого по нулевой, указывают на прерывания в соответствующих каналах. Разряд 0 указывает на прерывание в канале 0, разряд 1 - на прерывание в канале 1, и т.д.

47.8.4.2 Регистры StatusTfr, StatusBlock, StatusSrcTran, StatusDstTran, StatusErr

Регистры StatusTfr, StatusBlock, StatusSrcTran, StatusDstTran, StatusErr - маскируемые регистры прерываний. После маскирования события прерываний сохраняются в этих регистрах. См. Раздел «1.7 Прерывания».

Смещение	StatusTfr - 0x2e8 StatusBlock - 0x2f0 StatusSrcTran - 0x2f8 StatusDstTran - 0x300 StatusErr - 0x308
Тип	R
Состояние по сбросу	0x0
Разрядность	64

Таблица 47.16. Описание полей регистров StatusTfr, StatusBlock, StatusSrcTran, StatusDstTran, StatusErr

Разряды	Поле	Описание
63:8	Undefined	Зарезервировано.
7:0	STATUS	Разряды с седьмого по нулевой указывают на событие прерывания в соответствующих каналах после маскирования. Разряд 0 указывает на прерывание в канале 0, разряд 1 - на прерывание в канале 1, и т.д.

47.8.4.3 Регистры MaskTfr, MaskBlock, MaskSrcTran, MaskDstTran, MaskErr

Регистры MaskTfr, MaskBlock, MaskSrcTran, MaskDstTran, MaskErr - регистры масок прерываний. См. Раздел «1.7 Прерывания». Содержимое регистров Raw* маскируется регистрами Mask* и сохраняется в регистры Status*.

Смещение	MaskTfr - 0x310 MaskBlock - 0x318 MaskSrcTran - 0x320 MaskDstTran - 0x328 MaskErr - 0x330
Тип	RW
Состояние по сбросу	0x0
Разрядность	64

Таблица 47.17. Описание полей регистров MaskTfr, MaskBlock, MaskSrcTran, MaskDstTran, MaskErr

Разряды	Поле	Описание
63:8	Undefined	Зарезервировано.
15:8	INT_MASK_WE	Разрешение записи в соответствующие разряды поля INT_MASK. Например, запись 01X1(шестн.) в регистр MaskTfr установит 1 в нулевой разряд MaskTfr, не изменяя значения остальных разрядов.
7:0	INT_MASK	Запись единицы в разряды INT_MASK разрешает формирования прерывания для соответствующего Канала. Разряд 0 разрешает/запрещает прерывание в канале 0, разряд 1 разрешает/запрещает прерывание в канале 1, и т.д.

47.8.4.4 Регистры ClearTfr, ClearBlock, ClearSrcTran, ClearDstTran, ClearErr

Регистры ClearTfr, ClearBlock, ClearSrcTran, ClearDstTran, ClearErr - регистры сброса прерываний. См. Раздел «1.7 Прерывания».

Каждый разряд регистров Raw* и Status* очищается на одном и том же такте записью единицы в соответствующий разряд регистров Clear*. Запись нуля не оказывает влияния.

Регистры недоступны по чтению.

Смещение	ClearTfr - 0x338 ClearBlock - 0x340 ClearSrcTran - 0x348 ClearDstTran - 0x350 ClearErr - 0x358
Тип	W
Состояние по сбросу	0x0
Разрядность	64

Таблица 47.18. Описание полей регистров ClearTfr, ClearBlock, ClearSrcTran, ClearDstTran, ClearErr

Разряды	Поле	Описание
63:8	Undefined	Зарезервировано.
7:0	CLEAR	Разряды с седьмого по нулевой отвечают за сброс события прерывания в соответствующем канале. Разряд 0 сбрасывает прерывание в канале 0, разряд 1 - прерывание в канале 1, и т.д.

47.8.4.5 Регистр StatusInt

Регистр StatusInt (Status for each interrupt type) - Объединённый регистр прерываний. См. Раздел «Прерывания». Регистр доступен только по чтению.

Смещение	0x360
Тип	R
Состояние по сбросу	0x0
Разрядность	64

Таблица 47.19. Описание полей регистра StatusInt

Разряды	Поле	Описание
63:5	Undefined	Зарезервировано.
4	ERR	Взятое по ИЛИ содержимое регистра StatusErr.
3	DSTT	Взятое по ИЛИ содержимое регистра StatusDstTran.
2	SRCT	Взятое по ИЛИ содержимое регистра StatusSrcTran.
1	BLOCK	Взятое по ИЛИ содержимое регистра StatusBlock.
0	TFR	Взятое по ИЛИ содержимое регистра StatusTfr.

47.8.5 Регистры Интерфейса Запросов

В данном разделе описаны регистры *Программного Интерфейса Запросов*. Регистры *Программного Интерфейса Запросов* позволяют инициировать PDMA транзакции (Пакеты PDMA или *Одиночные обращения PDMA*), аналогично сигналам Аппаратных Интерфейсов Запросов. См. Раздел «*Интерфейсы Запросов*». Чтобы разрешить Программный Интерфейс Запросов Источника для *Канала x*, необходимо установить в единицу CFGx.HS_SEL_SRC. Чтобы разрешить *Программный Интерфейс Запросов Приёмника* для *Канала x*, необходимо установить в единицу CFGx.HS_SEL_DST.

47.8.5.1 Регистр ReqSrcReg

Регистр ReqSrcReg (Source Software Transaction Request Register) - Программный запрос передачи пакета Источника.

Каждому Каналу соответствует определённый разряд запроса в этом регистре. Включение запроса передачи пакета от источника инициирует чтение пакета контроллером PDMA от Источника. Если программный интерфейс запросов для источника канала N отключён, то соответствующие разряды этого регистра игнорируются.

Смещение	0x368
Тип	RW
Состояние по сбросу	0x0
Разрядность	64

Таблица 47.20. Описание полей регистра ReqSrcReg

Разряды	Поле	Описание
63:16	Undefined	Зарезервировано.
15:8	SRC_REQ_WE	Разрешение записи в разряды SRC_REQ. 0 - Запись запрещена. 1 - Запись разрешена. Поле доступно только по записи.
7:0	SRC_REQ	Запросы передачи пакета от источника. Разряды от 7 до нуля, соответствуют каналам от 7-го до нулевого. Запись в эти разряды происходит только с совместно установленными соответствующими разрядами поля SRC_REQ_WE, при этом канал должен быть включён через регистр ChEnReg.

47.8.5.2 Регистр ReqDstReg

Регистр ReqDstReg (Destination Software Transaction Request Register) - Программный запрос передачи пакета Приёмнику.

Каждому каналу соответствует определённый разряд запроса в этом регистре. Включение запроса передачи пакета Приёмнику инициирует передачу контроллером пакета приёмнику по шине AHB. Если программный интерфейс запросов для приёмника канала N отключён, то соответствующие разряды этого регистра игнорируются.

Смещение	0x370
Тип	RW
Состояние по сбросу	0x0
Разрядность	64

Таблица 47.21. Описание полей регистра ReqDstReg

Разряды	Поле	Описание
63:16	Undefined	Зарезервировано.
15:8	DST_REQ_WE	Разрешение записи в разряды DST_REQ. 0 - Запись запрещена. 1 - Запись разрешена. Поле доступно только по записи.

Разряды	Поле	Описание
7:0	DST_REQ	Запросы передачи пакета приёмнику. Разряды от 7 до нуля, соответствуют каналам от 7-го до нулевого. Запись в эти разряды происходит только, если установлены соответствующие разряды поля DST_REQ_WE.

47.8.5.3 Регистр SglReqSrcReg

Регистр SglReqSrcReg (Single Source Transaction Request Register) - Программный запрос одиночного чтения Источника.

Каждому каналу соответствует определённый разряд запроса в этом регистре. Включение запроса одиночного чтения источника инициирует одиночное обращения PDMA к Источнику. Если программный интерфейс запросов для источника канала N отключён, то соответствующие разряды этого регистра игнорируются.

Смещение	0x378
Тип	RW
Состояние по сбросу	0x0
Разрядность	64

Таблица 47.22. Описание полей регистра SglReqSrcReg

Разряды	Поле	Описание
63:16	Undefined	Зарезервировано.
15:8	SRC_SGLREQ_WE	Разрешение записи в разряды SRC_SGLREQ. 0 - Запись запрещена. 1 - Запись разрешена. Поле доступно только по записи.
7:0	SRC_SGLREQ	Запросы одиночных чтений Источника. Разряды от 7 до нуля, соответствуют каналам от 7-го до нулевого. Запись в эти разряды происходит только, если установлены соответствующие разряды поля SRC_SGLREQ_WE.

47.8.5.4 Регистр SglReqDstReg

Регистр SglReqDstReg (Single Destination Transaction Request Register) - Программный запрос одиночного обращения Приёмника.

Каждому каналу соответствует определённый разряд запроса в этом регистре. Включение запроса одиночной записи приёмника инициирует одиночное обращения PDMA к Приёмнику. Если программный интерфейс запросов для приёмника канала N отключён, то соответствующие разряды этого регистра игнорируются.

Смещение	0x380
Тип	RW
Состояние по сбросу	0x0
Разрядность	64

Таблица 47.23. Описание полей регистра SglReqDstRe

Разряды	Поле	Описание
63:16	Undefined	Зарезервировано.
15:8	DST_SGLREQ_WE	Разрешение записи в разряды DST_SGLREQ.

Разряды	Поле	Описание
		0 - Запись запрещена. 1 - Запись разрешена. Поле доступно только по записи.
7:0	DST_SGLREQ	Запросы одиночных обращений Приёмника. Разряды от 7 до нуля, соответствуют каналам от 7-го до нулевого. Запись в эти разряды происходит только, если установлены соответствующие разряды поля DST_SGLREQ_WE.

47.8.5.5 Регистр LstSrcReg

Регистр LstSrcReg (Last Source Transaction Request Register) - Последнее обращение Источника. Если для запроса (пакета или одиночного обращения) установлен этот признак, после обработки этого запроса PDMA завершит обработку Блока. Каждому каналу соответствует определённый разряд в этом регистре. Разряды этого регистра игнорируются, если программный интерфейс запросов для источника канала N отключён или если источник не управляет размером Блока.

Смещение	0x388
Тип	RW
Состояние по сбросу	0x0
Разрядность	64

Таблица 47.24. Описание полей регистра LstSrcReg

Разряды	Поле	Описание
63:16	Undefined	Зарезервировано.
15:8	LSTSRC_WE	Разрешение записи в разряды LSTSRC. 0 - Запись запрещена. 1 - Запись разрешена. Поле доступно только по записи.
7:0	LSTSRC	Признак последнего обращения (пакета или одиночного обращения) Источника. Разряды от 7 до нуля, соответствуют каналам от 7-го до нулевого. Запись в эти разряды происходит только, если установлены соответствующие разряды поля LSTSRC_WE. 0 - Обращение не является последним в Блоке. 1 - Последнее обращение в текущем Блоке.

47.8.5.6 Регистр LstDstReg

Регистр LstDstReg (Last Destination Transaction Request Register) - Последнее обращение к Приёмнику. Если для запроса (пакета или одиночного обращения) установлен этот признак, после обработки этого запроса PDMA завершит обработку Блока. Каждому каналу соответствует определённый разряд в этом регистре. Разряды этого регистра игнорируются, если программный интерфейс запросов для приёмника канала N отключён или если приёмник не управляет размером Блока.

Смещение	0x390
Тип	RW
Состояние по сбросу	0x0
Разрядность	64

Таблица 47.25. Описание полей регистра LstDstReg

Разряды	Поле	Описание
63:16	Undefined	Зарезервировано.
15:8	LSTDST_WE	Разрешение записи в разряды LSTDST. 0 - Запись запрещена. 1 - Запись разрешена. Поле доступно только по записи.
7:0	LSTDST	Признак последнего обращения (пакета или одиночного обращения) Приёмника. Разряды от 7 до нуля, соответствуют каналам от 7-го до нулевого. Запись в эти разряды происходит только, если установлены соответствующие разряды поля LSTDST_WE. 0 - Обращение не является последним в Блоке. 1 - Последнее обращение в текущем Блоке.

47.8.6 Регистры Конфигурации и Управления

47.8.6.1 Регистр DmaCfgReg

Регистр DmaCfgReg (DMA Configuration Register) - Конфигурационный регистр PDMA.

Регистр используется для глобального разрешения работы PDMA. Перед включением каналов необходимо установить глобальное разрешение работы.

Смещение	0x398
Тип	RW
Состояние по сбросу	0x0
Разрядность	64

Таблица 47.26. Описание полей регистра DmaCfgReg

Разряды	Поле	Описание
63:1	Undefined	Зарезервировано.
0	DMA_EN	Разрешения работы PDMA. 0 - PDMA отключён. 1 - PDMA включён. Если во время работы любого из Каналов записать в этот разряд 0, то при чтении он будет возвращать 1 до тех пор, пока PDMA не остановит активность на всех Каналах. После этого разряд будет возвращать 0.

47.8.6.2 Регистр ChEnReg

Регистр ChEnReg (Channel Enable Register) - Регистр включения каналов PDMA.

Регистр позволяет определять доступные (неактивные) Каналы, а также разрешать/запрещать работу необходимых каналов. Все разряды регистра очищаются в ноль, когда глобальный разряд включения PDMA (DmaCfgReg[0]) в нуле. Если глобальный разряд включения PDMA в нуле, записи в этот регистр будут игнорироваться и чтения всегда будут возвращать ноль.

Смещение	0x3A0
Тип	RW
Состояние по сбросу	0x0
Разрядность	64

Таблица 47.27. Описание полей регистра ChEnReg

Разряды	Поле	Описание
63:16	Undefined	Зарезервировано.
15:8	CH_EN_WE	Разрешения записи в разряды разрешения работы каналов.
7:0	CH_EN	Разрешение/Запрет работы канала. 0 - Запрет работы. 1 - Разрешение работы. Каждому каналу соответствует один разряд поля CH_EN. Разряды поля CH_EN аппаратно сбрасываются в ноль после завершения последней AHB транзакции к Приёмнику в PDMA Пересылке. Опрашивая эти разряды, поможет определить доступные для работы каналы. Запись в разряды этого поля возможна только, если установлены соответствующие разряды CH_EN_WE. Например, запись значения 01X1 (шестн.) в этот регистр вызовет установку единицы в разряде ChEnReg[0], а остальные разряды ChEnReg[7:1] оставит незатронутыми. См. 1.6 «Выключение канала до завершения пересылки».

47.8.6.3 Регистр DmaIdReg

Регистр DmaIdReg (DMA ID Register) - ID регистр PDMA.

Регистр доступен только по чтению. Возвращает аппаратно заданный идентификатор контроллера PDMA.

Смещение	0x3a8
Тип	R
Состояние по сбросу	0xeee ???
Разрядность	64

Таблица 47.28. Описание полей регистра DmaIdReg

Разряды	Поле	Описание
63:32	Undefined	Зарезервировано.
31:0	DMA_ID	Аппаратно заданный ID PDMA.

47.8.6.4 Регистр DmaTestReg

Регистр DmaTestReg (DMA Test Register) - Тестовый регистр PDMA.

Регистр используется для перевода ведомого порта AHB PDMA в тестовый режим. В тестовом режиме значение, записанное в регистр PDMA доступный по записи, совпадает со значением, которое возвращается последующими чтениями. В нормальном режиме значение, возвращаемое при чтении, зависит от состояния PDMA и может не совпадать с записанным.

Смещение	0x3b0
Тип	RW
Состояние по сбросу	0x0
Разрядность	64

Таблица 47.29. Описание полей регистра DmaTestReg

Разряды	Поле	Описание
63:1	Undefined	Зарезервировано.

Разряды	Поле	Описание
0	TEST_SLV_IF	Переводит конфигурационный (ведомый) порт AHB в тестовый режим. В тестовом режиме при чтении регистров, доступных по записи, всегда возвращается значение, установленное предшествующей записью. Разряд не разрешает запись в регистры доступные только по чтению. 0 - Нормальный режим. 1 - Тестовый режим.

47.8.7 Регистры Аппаратных Параметров

Регистры аппаратных параметров - доступные только по чтению регистры, содержащие информацию об аппаратных параметрах PDMA. О каждом из Каналов PDMA можно узнать следующую информацию:

Таблица 47.30. Аппаратные параметры Каналов PDMA

Параметр Канала x (x от 0 до 7)	Описание
Chx_FIFO_DEPTH[2:0]	Глубина FIFO канала в байтах. 0 = 8 байт 1 = 16 байт 2 = 32 байта 3 = 64 байта 4 = 128 байт 5 = 256 байт
CHx_SMS[2:0]	AHB порт Источника. 0 = AHB 1 1 = AHB 2 2 = AHB 3 3 = AHB 4 4 = Жёстко не задан
CHx_LMS[2:0]	AHB порт LLP. 0 = AHB 1 1 = AHB 2 2 = AHB 3 3 = AHB 4 4 = Жёстко не задан
CHx_DMS[2:0]	AHB порт Приёмника. 0 = AHB 1 1 = AHB 2 2 = AHB 3 3 = AHB 4 4 = Жёстко не задан
CHx_MAX_MULT_SIZE[2:0]	Максимальный размер Пакета PDMA. 0 = 4 1 = 8 2 = 16 3 = 32 4 = 64 5 = 128 6 = 256 7 = Зарезервировано
Chx_FC[1:0]	Тип управления размером Блока. 0 = только PDMA 1 = только Источник 2 = только Приёмник 3 = любое устройство

Параметр Канала x (x от 0 до 7)	Описание
	Если выбран тип управления 3, тогда управляющее устройство назначается программно.
CHx_HC_LLP	Поддержка цепочек Блоков. 1 - Регистр LLP всегда в нуле. Передачи цепочками блоков не поддерживаются. Информация о статусах Источника и Приёмника не обрабатывается (см. STATx и DSTATx), т. е информация о статусе не считывается из периферии и не записывается в блок LLI. Отключена возможность обратной записи.
CHx_CTL_WB_EN	Поддержка обратной записи. 1 - Поддержка обратной записи реализована (см. CTLx, SSTATx, DSTATx).
CHx_MULTI_BLK_EN	Поддержка многоблочных Пересылок. 0 - Поддерживаются только одноблочные Пересылки. 1 - Многоблочные Пересылки поддерживаются
CHx_LOCK_EN	Поддержка блокировок Канала и Шины. 0 - Поддержка блокировки Канала и Шины не реализована 1 - Поддержка блокировки Канала и Шины реализована
CHx_SRC_GAT_EN	Поддержка Сбора Источника. 0 - Сбор Источника не поддерживается 1 - Сбор Источника поддерживается
CHx_DST_SCA_EN	Поддержка Разброса Приёмника. 0 - Разброса Приёмника не поддерживается 1 - Разброса Приёмника поддерживается
CHx_STAT_SRC	Поддержка обработки статуса Источника. 0 - чтение и обратная запись статуса не поддерживается 1 - чтение и обратная запись статуса поддерживается
CHx_STAT_DST	Поддержка обработки статуса Приёмника. 0 - чтение и обратная запись статуса не поддерживается 1 - чтение и обратная запись статуса поддерживается
CHx_STW	Разрядность передач Источника. 0 = Не задана 1 = 8 2 = 16 3 = 32 4 = 64 5 = 128 6 = 256 7 = Зарезервировано
CHx_DTW	Разрядность передач Приёмника. 0 = Не задана 1 = 8 2 = 16 3 = 32 4 = 64 5 = 128 6 = 256 7 = Зарезервировано

47.8.7.1 Регистр DMA_COMP_PARAMS_6

Регистр DMA_COMP_PARAMS_6 – 6-ой регистр аппаратных параметров PDMA. Содержит информацию об аппаратной конфигурация канала 7.

Смещение	0x3c8
Тип	R

Смещение	0x3c8
Состояние по сбросу	Определяется настройками блока
Разрядность	64

Таблица 47.31. Описание полей регистра DMA_COMP_PARAMS_6

Разряды	Поле	Описание
63	–	Зарезервировано.
62:60	CH7_FIFO_DEPTH	Глубина FIFO канала в байтах.
59:57	CH7_SMS	AHB порт Источника.
56:54	CH7_LMS	AHB порт LLP.r
53:51	CH7_DMS	AHB порт Приёмника.
50:48	CH7_MAX_MULT_SIZE	Максимальный размер Пакета.
47:46	CH7_FC	Тип управления размером Блока.
45	CH7_HC_LLP	Поддержка цепочек Блоков.
44	CH7_CTL_WB_EN	Поддержка обратной записи.
43	CH7_MULTI_BLK_EN	Поддержка многоблочных Пересылок.
42	CH7_LOCK_EN	Поддержка блокировок Канала и Шины.
41	CH7_SRC_GAT_EN	Поддержка Сбора Источника.
40	CH7_DST_SCA_EN	Поддержка Разброса Приёмника.
39	CH7_STAT_SRC	Поддержка обработки статуса Источника.
38	CH7_STAT_DST	Поддержка обработки статуса Приёмника.
37:35	CH7_STW	Разрядность передач Источника.
34:32	CH7_DTW	Разрядность передач Приёмника.
31:0	–	Зарезервировано.

47.8.7.2 Регистр DMA_COMP_PARAMS_5

Регистр DMA_COMP_PARAMS_5 - 5-ый регистр аппаратных параметров PDMA. Содержит информацию об аппаратной конфигурация каналов 6 и 5.

Смещение	0x3d0
Тип	R
Состояние по сбросу	Определяется настройками блока
Разрядность	64

Таблица 47.32. Описание полей регистра DMA_COMP_PARAMS_5

Разряды	Поле	Описание
63	–	Зарезервировано.
62:60	CH5_FIFO_DEPTH	Глубина FIFO канала в байтах.
59:57	CH5_SMS	AHB порт Источника.
56:54	CH5_LMS	AHB порт LLP.
53:51	CH5_DMS	AHB порт Приёмника.
50:48	CH5_MAX_MULT_SIZE	Максимальный размер Пакета.
47:46	CH5_FC	Тип управления размером Блока.
45	CH5_HC_LLP	Поддержка цепочек Блоков.
44	CH5_CTL_WB_EN	Поддержка обратной записи.
43	CH5_MULTI_BLK_EN	Поддержка многоблочных Пересылок.
42	CH5_LOCK_EN	Поддержка блокировок Канала и Шины.
41	CH5_SRC_GAT_EN	Поддержка Сбора Источника.
40	CH5_DST_SCA_EN	Поддержка Разброса Приёмника.
39	CH5_STAT_SRC	Поддержка обработки статуса Источника.
38	CH5_STAT_DST	Поддержка обработки статуса Приёмника.
37:35	CH5_STW	Разрядность передач Источника.
34:32	CH5_DTW	Разрядность передач Приёмника.
31	–	Зарезервировано.
30:28	CH6_FIFO_DEPTH	Глубина FIFO канала в байтах.

Разряды	Поле	Описание
27:25	CH6_SMS	AHB порт Источника.
24:22	CH6_LMS	AHB порт LLP.
21:19	CH6_DMS	AHB порт Приёмника.
18:16	CH6_MAX_MULT_SIZE	Максимальный размер Пакета.
15:14	CH6_FC	Тип управления размером Блока.
13	CH6_HC_LLP	Поддержка цепочек Блоков.
12	CH6_CTL_WB_EN	Поддержка обратной записи.
11	CH6_MULTI_BLK_EN	Поддержка многоблочных Пересылок.
10	CH6_LOCK_EN	Поддержка блокировок Канала и Шины.
9	CH6_SRC_GAT_EN	Поддержка Сбора Источника.
8	CH6_DST_SCA_EN	Поддержка Разброса Приёмника.
7	CH6_STAT_SRC	Поддержка обработки статуса Источника.
6	CH6_STAT_DST	Поддержка обработки статуса Приёмника.
5:3	CH6_STW	Разрядность передач Источника.
2:0	CH6_DTW	Разрядность передач Приёмника.

47.8.7.3 Регистр DMA_COMP_PARAMS_4

Регистр DMA_COMP_PARAMS_4 - 4-ый регистр аппаратных параметров PDMA. Содержит информацию об аппаратной конфигурация каналов 4 и 3.

Смещение	0x3d8
Тип	R
Состояние по сбросу	Определяется настройками блока
Разрядность	64

Таблица 47.33. Описание полей регистра DMA_COMP_PARAMS_4

Разряды	Поле	Описание
63	–	Зарезервировано.
62:60	CH3_FIFO_DEPTH	Глубина FIFO канала в байтах.
59:57	CH3_SMS	AHB порт Источника.
56:54	CH3_LMS	AHB порт LLP.
53:51	CH3_DMS	AHB порт Приёмника.
50:48	CH3_MAX_MULT_SIZE	Максимальный размер Пакета.
47:46	CH3_FC	Тип управления размером Блока.
45	CH3_HC_LLP	Поддержка цепочек Блоков.
44	CH3_CTL_WB_EN	Поддержка обратной записи.
43	CH3_MULTI_BLK_EN	Поддержка многоблочных Пересылок.
42	CH3_LOCK_EN	Поддержка блокировок Канала и Шины.
41	CH3_SRC_GAT_EN	Поддержка Сбора Источника.
40	CH3_DST_SCA_EN	Поддержка Разброса Приёмника.
39	CH3_STAT_SRC	Поддержка обработки статуса Источника.
38	CH3_STAT_DST	Поддержка обработки статуса Приёмника.
37:35	CH3_STW	Разрядность передач Источника.
34:32	CH3_DTW	Разрядность передач Приёмника.
31	–	Зарезервировано.
30:28	CH4_FIFO_DEPTH	Глубина FIFO канала в байтах.
27:25	CH4_SMS	AHB порт Источника.
24:22	CH4_LMS	AHB порт LLP.
21:19	CH4_DMS	AHB порт Приёмника.
18:16	CH4_MAX_MULT_SIZE	Максимальный размер Пакета.
15:14	CH4_FC	Тип управления размером Блока.
13	CH4_HC_LLP	Поддержка цепочек Блоков.
12	CH4_CTL_WB_EN	Поддержка обратной записи.
11	CH4_MULTI_BLK_EN	Поддержка многоблочных Пересылок.

Разряды	Поле	Описание
10	CH4_LOCK_EN	Поддержка блокировок Канала и Шины.
9	CH4_SRC_GAT_EN	Поддержка Сбора Источника.
8	CH4_DST_SCA_EN	Поддержка Разброса Приёмника.
7	CH4_STAT_SRC	Поддержка обработки статуса Источника.
6	CH4_STAT_DST	Поддержка обработки статуса Приёмника.
5:3	CH4_STW	Разрядность передач Источника.
2:0	CH4_DTW	Разрядность передач Приёмника.

47.8.7.4 Регистр DMA_COMP_PARAMS_3

Регистр DMA_COMP_PARAMS_3 - 3-ий регистр аппаратных параметров PDMA. Содержит информацию об аппаратной конфигурация каналов 2 и 1.

Смещение	0x3e0
Тип	R
Состояние по сбросу	Определяется настройками блока
Разрядность	64

Таблица 47.34. Описание полей регистра DMA_COMP_PARAMS_3

Разряды	Поле	Описание
63	–	Зарезервировано.
62:60	CH1_FIFO_DEPTH	Глубина FIFO канала в байтах.
59:57	CH1_SMS	AHB порт Источника.
56:54	CH1_LMS	AHB порт LLP.
53:51	CH1_DMS	AHB порт Приёмника.
50:48	CH1_MAX_MULT_SIZE	Максимальный размер Пакета.
47:46	CH1_FC	Тип управления размером Блока.
45	CH1_HC_LLP	Поддержка цепочек Блоков.
44	CH1_CTL_WB_EN	Поддержка обратной записи.
43	CH1_MULTI_BLK_EN	Поддержка многоблочных Пересылок.
42	CH1_LOCK_EN	Поддержка блокировок Канала и Шины.
41	CH1_SRC_GAT_EN	Поддержка Сбора Источника.
40	CH1_DST_SCA_EN	Поддержка Разброса Приёмника.
39	CH1_STAT_SRC	Поддержка обработки статуса Источника.
38	CH1_STAT_DST	Поддержка обработки статуса Приёмника.
37:35	CH1_STW	Разрядность передач Источника.
34:32	CH1_DTW	Разрядность передач Приёмника.
31	–	Зарезервировано.
30:28	CH2_FIFO_DEPTH	Глубина FIFO канала в байтах.
27:25	CH2_SMS	AHB порт Источника.
24:22	CH2_LMS	AHB порт LLP.
21:19	CH2_DMS	AHB порт Приёмника.
18:16	CH2_MAX_MULT_SIZE	Максимальный размер Пакета.
15:14	CH2_FC	Тип управления размером Блока.
13	CH2_HC_LLP	Поддержка цепочек Блоков.
12	CH2_CTL_WB_EN	Поддержка обратной записи.
11	CH2_MULTI_BLK_EN	Поддержка многоблочных Пересылок.
10	CH2_LOCK_EN	Поддержка блокировок Канала и Шины.
9	CH2_SRC_GAT_EN	Поддержка Сбора Источника.
8	CH2_DST_SCA_EN	Поддержка Разброса Приёмника.
7	CH2_STAT_SRC	Поддержка обработки статуса Источника.
6	CH2_STAT_DST	Поддержка обработки статуса Приёмника.
5:3	CH2_STW	Разрядность передач Источника.
2:0	CH2_DTW	Разрядность передач Приёмника.

47.8.7.5 Регистр DMA_COMP_PARAMS_2

Регистр DMA_COMP_PARAMS_2 - 2-ой регистр аппаратных параметров PDMA. Содержит информацию об аппаратной конфигурация канала 0 и типах многоблочных пересылок для всех каналов.

Смещение	0x3e8
Тип	R
Состояние по сбросу	Определяется настройками блока
Разрядность	64

Таблица 47.35. Описание полей регистра DMA_COMP_PARAMS_2

Разряды	Поле	Описание
63:60	CH7_MULTI_BLK_TYPE	Тип многоблочных Пересылок канала 7. 0 = NO_HARDCODE Любые типы. 1 = CONT_RELOAD SARx - смежный. DAR и CTL - автоповтор. 2 = RELOAD_CONT SARx и CTLx - автоповтор. DARx - смежный. 3 = RELOAD_RELOAD SARx, DARx, и CTLx - автоповтор. 4 = CONT_LLP SARx - смежный. DARx, CTLx, и LLPx загружаются из блоков LLI. 5 = RELOAD_LLP SARx - автоповтор. DARx, CTLx, и LLPx загружаются из блоков LLI. 6 = LLP_CONT SARx, CTLx, и LLPx загружаются из блоков LLI. DARx - смежный. 7 = LLP_RELOAD SARx, CTLx, и LLPx загружаются из блоков LLI. DARx - автоповтор. 8 = LLP_LLP SARx, DARx, CTLx, и LLPx загружаются из блоков LLI цепочки Блоков.
59:56	CH6_MULTI_BLK_TYPE	Тип многоблочных Пересылок канала 6.
55:52	CH5_MULTI_BLK_TYPE	Тип многоблочных Пересылок канала 5.
51:48	CH4_MULTI_BLK_TYPE	Тип многоблочных Пересылок канала 4.
47:44	CH3_MULTI_BLK_TYPE	Тип многоблочных Пересылок канала 3.
43:40	CH2_MULTI_BLK_TYPE	Тип многоблочных Пересылок канала 2.
39:36	CH1_MULTI_BLK_TYPE	Тип многоблочных Пересылок канала 1.
35:32	CH0_MULTI_BLK_TYPE	Тип многоблочных Пересылок канала 0.
31	–	Зарезервировано.
30:28	CH0_FIFO_DEPTH	Глубина FIFO канала в байтах.
27:25	CH0_SMS	AHB порт Источника.
24:22	CH0_LMS	AHB порт LLP.
21:19	CH0_DMS	AHB порт Приёмника.
18:16	CH0_MAX_MULT_SIZE	Максимальный размер Пакета.
15:14	CH0_FC	Тип управления размером Блока.
13	CH_HC_LLP	Поддержка цепочек Блоков.
12	CH0_CTL_WB_EN	Поддержка обратной записи.
11	CH0_MULTI_BLK_EN	Поддержка многоблочных Пересылок.
10	CH0_LOCK_EN	Поддержка блокировок Канала и Шины.

Разряды	Поле	Описание
9	CH0_SRC_GAT_EN	Поддержка Сбора Источника.
8	CH0_DST_SCA_EN	Поддержка Разброса Приёмника.
7	CH0_STAT_SRC	Поддержка обработки статуса Источника.
6	CH0_STAT_DST	Поддержка обработки статуса Приёмника.
5:3	CH0_STW	Разрядность передач Источника.
2:0	CH0_DTW	Разрядность передач Приёмника.

47.8.7.6 Регистр DMA_COMP_PARAMS_1

Регистр DMA_COMP_PARAMS_1 - 1-ый регистр аппаратных параметров PDMA.

Смещение	0x3f0
Тип	R
Состояние по сбросу	Определяется настройками блока
Разрядность	64

Таблица 47.36. Описание полей регистра DMA_COMP_PARAMS_1

Разряды	Поле	Описание
63:62	–	Зарезервировано.
61	STATIC_ENDIAN_SELECT	Способ задания порядка байтов в обменах AHB. 0 - Для каждого порта AHB порядок байтов определяется индивидуально отдельными аппаратными входами PDMA. 1 - Порядок байтов для всех портов AHB (как ведомых, так и ведущих) одинаков и жёстко задан при конфигурации.
60	ADD_ENCODED_PARAM	Определяет наличие регистров аппаратных параметров от 6-го до 1-го (DMA_COMP_PARAMS_..).
59:55	NUM_HS_INT	Количество аппаратных интерфейсов запросов (от 0 до 16).
54:53	M4_HDATA_WIDTH	Разрядность ведущего AHB порта 4. 0 - 32 разряда 1 - 64 разряда 2 - 128 разрядов 3 - 256 разрядов
52:51	M3_HDATA_WIDTH	Разрядность ведущего AHB порта 3.
50:49	M2_HDATA_WIDTH	Разрядность ведущего AHB порта 2.
48:47	M1_HDATA_WIDTH	Разрядность ведущего AHB порта 1.
46:45	S_HDATA_WIDTH	Разрядность конфигурационного AHB порта.
44:43	NUM_MASTER_INT	Количество ведущих портов AHB. 0 - 1 ведущий ... 3 - 4 ведущих
42:40	NUM_CHANNELS	Количество Каналов. 0 - 1 канал ... 7 - 8 каналов
39:36	–	Зарезервировано.
35	MABRST	Разрешение программного ограничения размера пакетов AHB. 0 - Максимальный размер пакета AHB определяется размером FIFO Канала. PDMA может заполнять и очищать FIFO за одно пакетное обращение. 1 - Максимальный размер пакета AHB определяется настройками регистров Канала.
34:33	INTR_IO	Аппаратно заданный тип сигналов прерываний: 0 - ALL Для каждого канала выведены все 5 возможных прерываний (любому прерыванию каждого из каналов соответствует отдельный сигнал).

Разряды	Поле	Описание
		1 - TYPE Выведены 5 сигналов прерываний согласно типу (каждый из сигналов указывает на возникновение прерывание заданного типа в любом из каналов). Также выведен объединённый сигнал прерываний (указывает на наличие любого прерывания в любом канале). 2 - COMBINED Выведен только объединённый сигнал прерываний (указывает на наличие любого прерывания в любом канале). 3 - зарезервировано
32	BIG_ENDIAN	Порядок байтов в обменах AHB. 0 - Сначала передаются младшие байты (little-endian) 1 - Сначала передаются старшие байты (big-endian) Поле работает только при установленном в единицу поле STATIC_ENDIAN_SELECT.
31:28	CH7_MAX_BLK_SIZE	Максимальный размер Блока канала 7. Определяет максимально количество обращений разрядностью передач источника в Блоке. Если размером Блока управляет PDMA, разрешается настраивать размер Блока, не превышающий значение этого параметра. Если размером Блока управляет периферия, то размер Блока может быть больше значения этого параметра. Параметр ограничивает размер границы Сбора/Разброса. 0x0 = 3 0x1 = 7 0x2 =15 0x3 = 31 0x4 = 63 0x5 = 127 0x6 = 255 0x7 = 511 0x8 = 1023 0x9 = 2047 0xa = 4095
27:24	CH6_MAX_BLK_SIZE	Максимальный размер Блока канала 6.
23:20	CH5_MAX_BLK_SIZE	Максимальный размер Блока канала 5.
19:16	CH4_MAX_BLK_SIZE	Максимальный размер Блока канала 4.
15:12	CH3_MAX_BLK_SIZE	Максимальный размер Блока канала 3.
11:8	CH2_MAX_BLK_SIZE	Максимальный размер Блока канала 2.
7:4	CH1_MAX_BLK_SIZE	Максимальный размер Блока канала 1.
3:0	CH0_MAX_BLK_SIZE	Максимальный размер Блока канала 0.

47.8.8 Регистр DMA_Component_ID_Register

Регистр DMA_Component_ID_Register (DMA Component ID Register) - ID регистр компонента PDMA.

Регистр доступен только по чтению. Регистр позволяет определить тип и версию компонента Designware.

Смещение	0x3f8
Тип	R
Состояние по сбросу	3231_372A__4457_1110
Разрядность	64

Таблица 47.37. Описание полей регистра DMA_Component_ID_Register

Разряды	Поле	Описание
63:32	DMA_COMP_VERSION	Версия компонента.
31:0	DMA_COMP_TYPE	Тип компонента Designware. Значение по сбросу - 0x44571110.

47.9 Настройки компонентов PDMA0, PDMA1, PDMA2

Таблица 47.38. Конфигурационные параметры PDMA0, PDMA1, PDMA2

Параметр	PDMA0	PDMA1	PDMA2
USE_FOUNDATION	1	1	1
DMAH_NUM_MASTER_INT	3	3	4
DMAH_NUM_CHANNELS	8	8	8
DMAH_NUM_HS_INT	12	13	2
DMAH_WLAST_EN	0	0	0
DMAH_ID_NUM	32'h2	32'h2	32'h2
DMAH_INTR_POL	1	1	1
DMAH_INTR_IO	2	2	2
DMAH_MABRST	1	1	1
DMAH_INCR_BURSTS	0	0	0
DMAH_RETURN_ERR_RESP	1	1	1
DMAH_ADD_ENCODED_PARAMS	1'h1	1'h1	1'h1
DMAH_REMOVE_PIPELINING	0	0	0
DMAH_STATIC_ENDIAN_SELECT	1	1	1
DMAH_BIG_ENDIAN	0	0	0
DMAH_BE32_SELECTION_PIN_EN	0	0	0
DMAH_LLI_ENDIAN_SELECTION_PIN_EN	0	0	0
DMAH_S_HDATA_WIDTH	32	32	32
DMAH_LP_EN	0	0	0
DMAH_CH_LP_EN	0	0	0
DMAH_LP_TIMEOUT_WIDTH	4	4	4
DMAH_HC_LP_TIMEOUT_VALUE	0	0	0
DMAH_LP_TIMEOUT_VALUE	32'd8	32'd8	32'd8
DMAH_M1_AHB_LITE	0	0	1

Параметр	PDMA0	PDMA1	PDMA2
DMAH_M1_HDATA_WIDTH	32	32	32
DMAH_M2_AHB_LITE	1	1	1
DMAH_M2_HDATA_WIDTH	32	32	64
DMAH_M3_AHB_LITE	1	1	1
DMAH_M3_HDATA_WIDTH	32	32	64
DMAH_M4_AHB_LITE	0	0	1
DMAH_M4_HDATA_WIDTH	32	32	64
DMAH_CH0_FIFO_DEPTH	256	256	256
DMAH_CH0_MAX_MULT_SIZE	256	256	256
DMAH_CH0_MAX_BLK_SIZE	4095	4095	4095
DMAH_CH0_FC	0	0	0
DMAH_CH0_LOCK_EN	0	0	0
DMAH_CH0_SMS	4	4	4
DMAH_CH0_DMS	4	4	4
DMAH_CH0_STW	0	0	0
DMAH_CH0_DTW	0	0	0
DMAH_CH0_SRC_NON_OK	1	1	1
DMAH_CH0_DST_NON_OK	1	1	1
DMAH_CH0_MULTI_BLK_EN	1	1	1
DMAH_CH0_HC_LLP	0	0	0
DMAH_CH0_MULTI_BLK_TYPE	0	0	0
DMAH_CH0_STAT_SRC	1	1	1
DMAH_CH0_STAT_DST	1	1	1
DMAH_CH0_LMS	4	4	4
DMAH_CH0_LLP_NON_OK	1	1	1
DMAH_CH0_SRC_GAT_EN	1	1	1
DMAH_CH0_DST_SCA_EN	1	1	1
DMAH_CH0_CTL_WB_EN	1	1	1
DMAH_CH1_FIFO_DEPTH	256	256	256
DMAH_CH1_MAX_MULT_SIZE	256	256	256

Параметр	PDMA0	PDMA1	PDMA2
DMAH_CH1_MAX_BLK_SIZE	4095	4095	4095
DMAH_CH1_FC	0	0	0
DMAH_CH1_LOCK_EN	0	0	0
DMAH_CH1_SMS	4	4	4
DMAH_CH1_DMS	4	4	4
DMAH_CH1_STW	0	0	0
DMAH_CH1_DTW	0	0	0
DMAH_CH1_SRC_NON_OK	1	1	1
DMAH_CH1_DST_NON_OK	1	1	1
DMAH_CH1_MULTI_BLK_EN	1	1	1
DMAH_CH1_MULTI_BLK_TYPE	0	0	0
DMAH_CH1_HC_LLP	0	0	0
DMAH_CH1_STAT_SRC	1	1	1
DMAH_CH1_STAT_DST	1	1	1
DMAH_CH1_LMS	4	4	4
DMAH_CH1_LLP_NON_OK	1	1	1
DMAH_CH1_SRC_GAT_EN	1	1	1
DMAH_CH1_DST_SCA_EN	1	1	1
DMAH_CH1_CTL_WB_EN	1	1	1
DMAH_CH2_FIFO_DEPTH	256	256	256
DMAH_CH2_MAX_MULT_SIZE	256	256	256
DMAH_CH2_MAX_BLK_SIZE	4095	4095	4095
DMAH_CH2_FC	0	0	0
DMAH_CH2_LOCK_EN	0	0	0
DMAH_CH2_SMS	4	4	4
DMAH_CH2_DMS	4	4	4
DMAH_CH2_STW	0	0	0
DMAH_CH2_DTW	0	0	0
DMAH_CH2_SRC_NON_OK	1	1	1
DMAH_CH2_DST_NON_OK	1	1	1

Параметр	PDMA0	PDMA1	PDMA2
DMAH_CH2_MULTI_BLK_EN	1	1	1
DMAH_CH2_MULTI_BLK_TYPE	0	0	0
DMAH_CH2_HC_LLP	0	0	0
DMAH_CH2_STAT_SRC	1	1	1
DMAH_CH2_STAT_DST	1	1	1
DMAH_CH2_LMS	4	4	4
DMAH_CH2_LLP_NON_OK	1	1	1
DMAH_CH2_SRC_GAT_EN	1	1	1
DMAH_CH2_DST_SCA_EN	1	1	1
DMAH_CH2_CTL_WB_EN	1	1	1
DMAH_CH3_FIFO_DEPTH	256	256	256
DMAH_CH3_MAX_MULT_SIZE	256	256	256
DMAH_CH3_MAX_BLK_SIZE	4095	4095	4095
DMAH_CH3_FC	0	0	0
DMAH_CH3_LOCK_EN	0	0	0
DMAH_CH3_SMS	4	4	4
DMAH_CH3_DMS	4	4	4
DMAH_CH3_STW	0	0	0
DMAH_CH3_DTW	0	0	0
DMAH_CH3_SRC_NON_OK	1	1	1
DMAH_CH3_DST_NON_OK	1	1	1
DMAH_CH3_MULTI_BLK_EN	1	1	1
DMAH_CH3_MULTI_BLK_TYPE	0	0	0
DMAH_CH3_HC_LLP	0	0	0
DMAH_CH3_STAT_SRC	1	1	1
DMAH_CH3_STAT_DST	1	1	1
DMAH_CH3_LMS	4	4	4
DMAH_CH3_LLP_NON_OK	1	1	1
DMAH_CH3_SRC_GAT_EN	1	1	1
DMAH_CH3_DST_SCA_EN	1	1	1

Параметр	PDMA0	PDMA1	PDMA2
DMAH_CH3_CTL_WB_EN	1	1	1
DMAH_CH4_FIFO_DEPTH	256	256	256
DMAH_CH4_MAX_MULT_SIZE	256	256	256
DMAH_CH4_MAX_BLK_SIZE	4095	4095	4095
DMAH_CH4_FC	0	0	0
DMAH_CH4_LOCK_EN	0	0	0
DMAH_CH4_SMS	4	4	4
DMAH_CH4_DMS	4	4	4
DMAH_CH4_STW	0	0	0
DMAH_CH4_DTW	0	0	0
DMAH_CH4_SRC_NON_OK	1	1	1
DMAH_CH4_DST_NON_OK	1	1	1
DMAH_CH4_MULTI_BLK_EN	1	1	1
DMAH_CH4_MULTI_BLK_TYPE	0	0	0
DMAH_CH4_HC_LLP	0	0	0
DMAH_CH4_STAT_SRC	1	1	1
DMAH_CH4_STAT_DST	1	1	1
DMAH_CH4_LMS	4	4	4
DMAH_CH4_LLP_NON_OK	1	1	1
DMAH_CH4_SRC_GAT_EN	1	1	1
DMAH_CH4_DST_SCA_EN	1	1	1
DMAH_CH4_CTL_WB_EN	1	1	1
DMAH_CH5_FIFO_DEPTH	256	256	256
DMAH_CH5_MAX_MULT_SIZE	256	256	256
DMAH_CH5_MAX_BLK_SIZE	4095	4095	4095
DMAH_CH5_FC	0	0	0
DMAH_CH5_LOCK_EN	0	0	0
DMAH_CH5_SMS	4	4	4
DMAH_CH5_DMS	4	4	4
DMAH_CH5_STW	0	0	0

Параметр	PDMA0	PDMA1	PDMA2
DMAH_CH5_DTW	0	0	0
DMAH_CH5_SRC_NON_OK	1	1	1
DMAH_CH5_DST_NON_OK	1	1	1
DMAH_CH5_MULTI_BLK_EN	1	1	1
DMAH_CH5_MULTI_BLK_TYPE	0	0	0
DMAH_CH5_HC_LLP	0	0	0
DMAH_CH5_STAT_SRC	1	1	1
DMAH_CH5_STAT_DST	1	1	1
DMAH_CH5_LMS	4	4	4
DMAH_CH5_LLP_NON_OK	1	1	1
DMAH_CH5_SRC_GAT_EN	1	1	1
DMAH_CH5_DST_SCA_EN	1	1	1
DMAH_CH5_CTL_WB_EN	1	1	1
DMAH_CH6_FIFO_DEPTH	256	256	256
DMAH_CH6_MAX_MULT_SIZE	256	256	256
DMAH_CH6_MAX_BLK_SIZE	4095	4095	4095
DMAH_CH6_FC	0	0	0
DMAH_CH6_LOCK_EN	0	0	0
DMAH_CH6_SMS	4	4	4
DMAH_CH6_DMS	4	4	4
DMAH_CH6_STW	0	0	0
DMAH_CH6_DTW	0	0	0
DMAH_CH6_SRC_NON_OK	1	1	1
DMAH_CH6_DST_NON_OK	1	1	1
DMAH_CH6_MULTI_BLK_EN	1	1	1
DMAH_CH6_MULTI_BLK_TYPE	0	0	0
DMAH_CH6_HC_LLP	0	0	0
DMAH_CH6_STAT_SRC	1	1	1
DMAH_CH6_STAT_DST	1	1	1
DMAH_CH6_LMS	4	4	4

Параметр	PDMA0	PDMA1	PDMA2
DMAH_CH6_LLP_NON_OK	1	1	1
DMAH_CH6_SRC_GAT_EN	1	1	1
DMAH_CH6_DST_SCA_EN	1	1	1
DMAH_CH6_CTL_WB_EN	1	1	1
DMAH_CH7_FIFO_DEPTH	256	256	256
DMAH_CH7_MAX_MULT_SIZE	256	256	256
DMAH_CH7_MAX_BLK_SIZE	4095	4095	4095
DMAH_CH7_FC	0	0	0
DMAH_CH7_LOCK_EN	0	0	0
DMAH_CH7_SMS	4	4	4
DMAH_CH7_DMS	4	4	4
DMAH_CH7_STW	0	0	0
DMAH_CH7_DTW	0	0	0
DMAH_CH7_SRC_NON_OK	1	1	1
DMAH_CH7_DST_NON_OK	1	1	1
DMAH_CH7_MULTI_BLK_EN	1	1	1
DMAH_CH7_MULTI_BLK_TYPE	0	0	0
DMAH_CH7_HC_LLP	0	0	0
DMAH_CH7_STAT_SRC	1	1	1
DMAH_CH7_STAT_DST	1	1	1
DMAH_CH7_LMS	4	4	4
DMAH_CH7_LLP_NON_OK	1	1	1
DMAH_CH7_SRC_GAT_EN	1	1	1
DMAH_CH7_DST_SCA_EN	1	1	1
DMAH_CH7_CTL_WB_EN	1	1	1
DMAH_REVERSE_WB_OVERRIDE	0	0	0

48. БЛОК НАКРИСТАЛЬНЫХ ДАТЧИКОВ (SENSORS)

48.1 Общая информация

Микросхема содержит 4 пары накристальных датчиков температуры и напряжения, которые расположены в областях кристалла микросхемы представляющих наибольший интерес.

Для датчиков температуры - это области возле блоков:

- GPU;
- CPU;
- DSP0;
- На краю кристалла в служебной подсистеме SERVICE.

Для датчиков напряжения – это области возле блоков:

- GPU (напряжение MVDD);
- CPU (напряжение AVDD);
- DSP0 (напряжение SVDD);
- На краю кристалла в служебной подсистеме SERVICE (напряжение CVDD).

Все датчики управляются с помощью единого контроллера PVT CTR, являющегося частью служебной подсистемы и доверенного контура. Контроллер имеет единый интерфейс и набор регистров, позволяющий считывать значения датчиков, управлять их настройкой, и генерировать события и прерывания в случае достижения пороговых значений.

Распределение датчиков по каналам контроллера PVT CTR показано в Таблица 48.1.

Таблица 48.1. Каналы датчиков температуры и напряжения PVT контроллера

Канал	Область, подсистема
ts[0], vm[0]	CPU, cpu_subs
ts[1], vm[1]	DSP0, sdr_subs
ts[2], vm[2]	GPU, media_subs
ts[3], vm[3]	die edge, service_subs

Управление и настройка датчиков осуществляется с помощью драйверов ОС, поставляемых с микросхемой.

49. СИСТЕМА ОТЛАДКИ МИКРОСХЕМЫ (DEBUG)

49.1 Общая информация

На следующем рисунке показана структурная схема системы отладки:

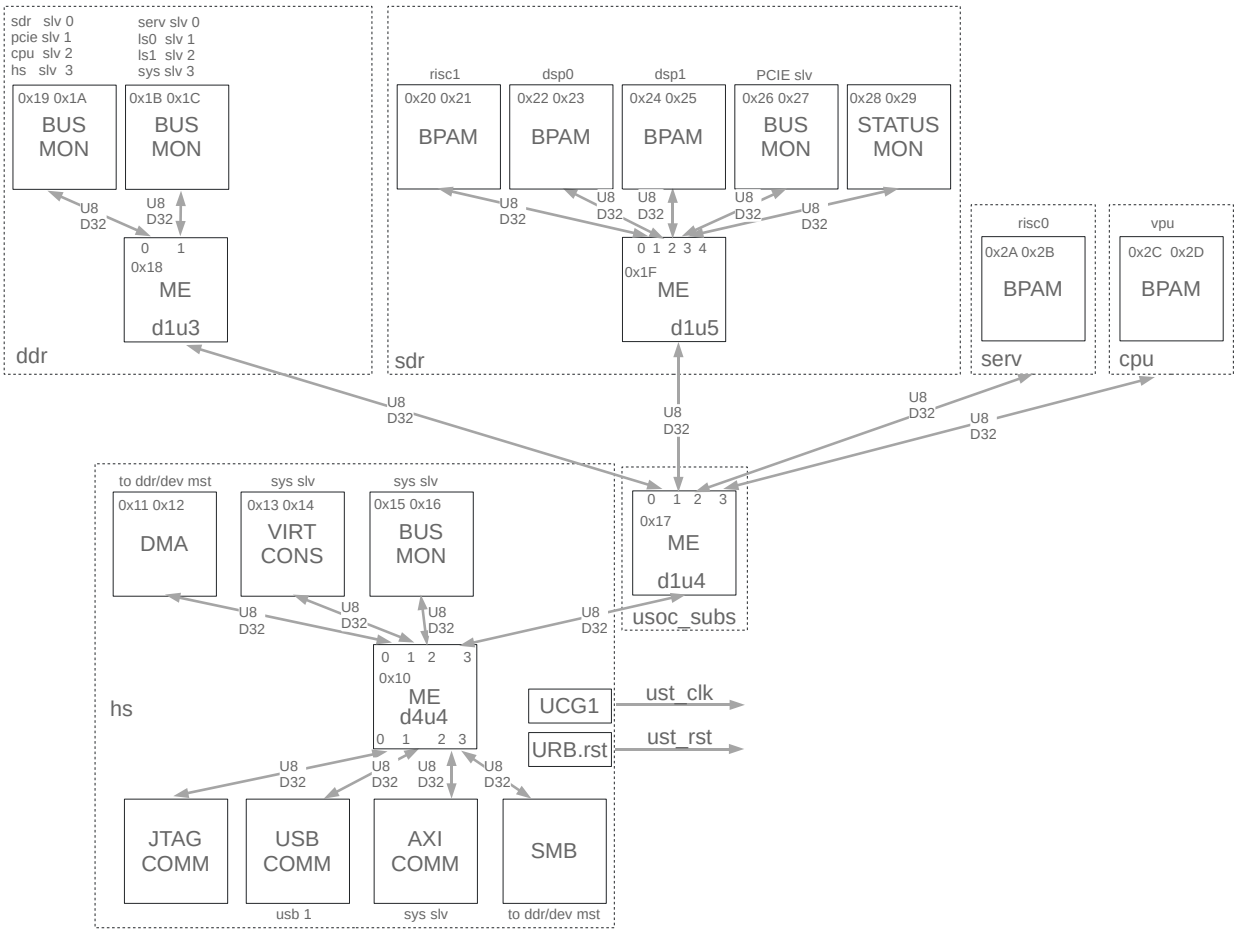


Рисунок 49.1. Структурная схема системы отладки

Основные свойства и назначение системы отладки:

- Подключение отладчика по USB2.0 (USB COMM)
- Подключение отладчика по JTAG (JTAG COMM)
- Доступ отладчика к системной памяти микросхемы, т.е. к памяти и регистрам устройств (DMA)
- Управление процессорными ядрами под отладчиком (блоки BPAM)
- Настройка и управление системой отладки со стороны процессорных ядер (AXI COMM)
- Виртуальная консоль, отладка в стиле printf (VIRT CONS)
- Сохранение отладочной информации в память (SMB)
- Оценка производительности, профилирование (блоки BUS MON и BPAM).

- Наблюдение, фильтрации обращений в интересующие области памяти (блоки BUS MON). Мониторы встроены почти во все основные слейвы шин доступа к памяти микросхемы (DDR, PCIe, NAND, SDMMC).
- Обмен событиями отладки между компонентам. Все компоненты соединяются при помощи блоков ME.

49.2 Тактирование и сброс

Все компоненты системы отладки тактируются от одной частоты ust_clk, которая формируется в блоке HSPERIPH.UCG1. Чтобы работать с отладкой предварительно нужно включить частоту отладки (см. регистры HSPERIPH.UCG1). Частота включается либо автоматически (см. описание ниже), либо с помощью ПО.

Сигнал ust_rst - общий сброс всей системы отладки, управляется через поле dbg регистра HSPERIPH.URB.rst. Сброс снимается либо автоматически (см. описание ниже), либо с помощью ПО.

Кроме общего сброса компоненты отладки находятся в состоянии сброса при дополнительных условиях (определяется настройками регистров SERV SUBS, см. описание ниже):

Таблица 49.1. Дополнительные условия сброса компонентов отладки

Компоненты	Дополнительные условие сброса
Подсистема ddr	dbg_disable | ~dbg_dbgen_ddr_i
Подсистема sdr	dbg_disable | ~dbg_dbgen_sdr_i
Подсистема serv	dbg_disable | ~dbg_dbgen_serv_i
Подсистема cpu	dbg_disable | ~dbg_dbgen_cpu_i
USB и JTAG COMM	dbg_disable | ~dbg_dbgen_usb_jtag_i
AXI COMM	dbg_disable | ~dbg_dbgen_acom_i
VIRT CONS	dbg_disable | ~dbg_dbgen_vc_i
DMA и SMB	dbg_disable | ~dbg_dbgen_dma_smb_i

49.3 Включение отладки

На следующем рисунке показаны основные сигналы отвечающие за включение отладки:

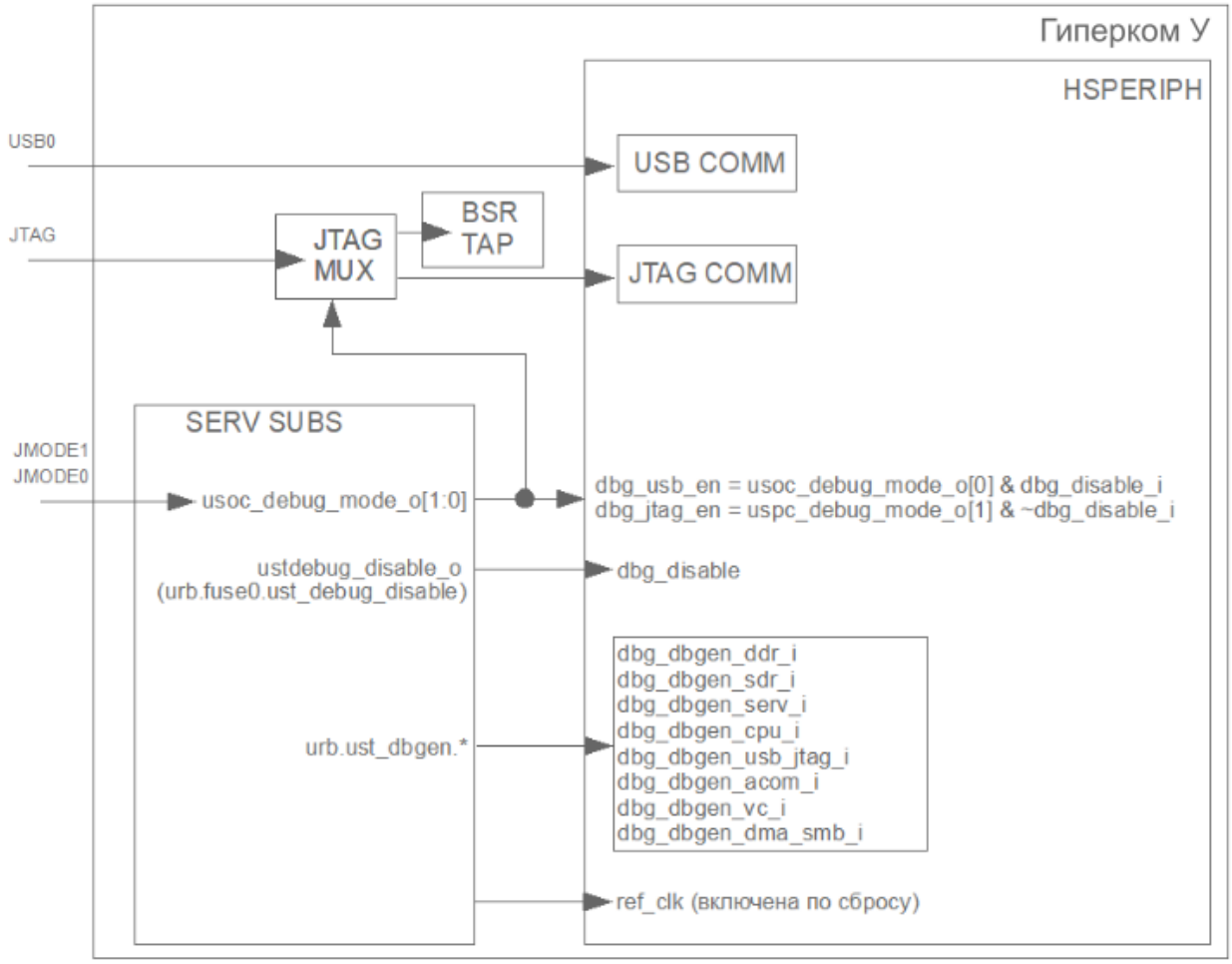


Рисунок 49.2. Основные сигналы включения отладки

При необходимости отладка отключается полностью через fuse-регистр подсистемы service_subs – urb.fuse0.ust_debug_disable.

Частично ограничить возможности отладки можно через обычный регистр подсистемы service_subs – urb.ust_dbgen.

Предусматривается два режима включения отладки – автоматический (virgin state) и ручной.

Вывод системы отладки из сброса рекомендуется производить на пониженной частоте, чтобы обеспечить синхронное распространение сигнала сброса до каждого блока системы отладки.

49.3.1 Автоматический режим

При использовании внешнего отладчика (USB, JTAG) перед выводом микросхемы из сброса необходимо установить внешние сигналы микросхемы, отвечающие за формирование внутренних сигналов включения dbg_usb_en, dbg_jtag_en (см. Рисунок 49.2). Запуск производится в режиме “virgin state”.

При установленных сигналах dbg_usb_en, dbg_jtag_en частота отладки включается автоматически после выхода из сброса. Дополнительно при установленном dbg_usb_en автоматически производится подключение USB COMM к PHY USB0 и инициализацией настроек USB phy на частоту 25MHz (источник - входной порт CLK_125MHz).

49.3.2 Ручной режим

В ручном режиме включение частот, разрешений отладки, настройка PHY USB полностью управляется накристальным ПО. В этом режиме подключение внешнего отладчика невозможно без участия накристального ПО.

49.4 AXI COMM

На следующем рисунке показана структурная схема блока AXI COMM:

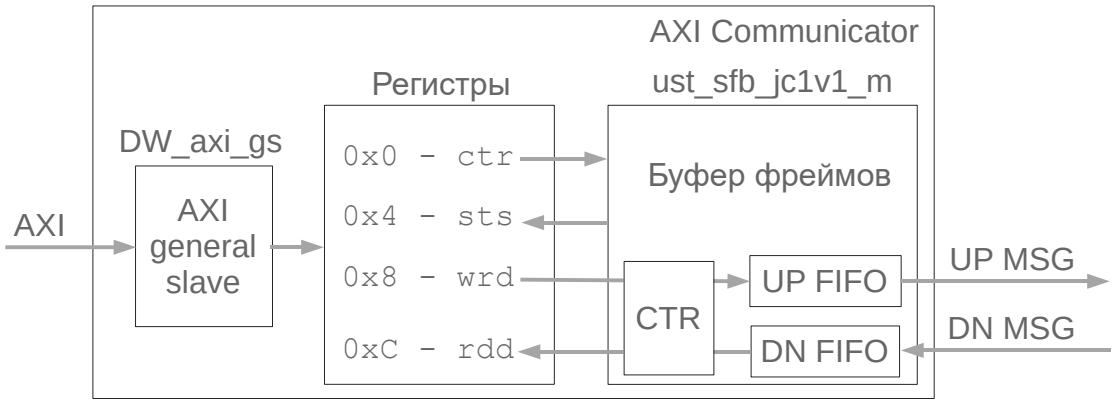


Рисунок 49.3. Структурная схема AXI COMM

За основу блока взят компонент ust_sfb_jc1v1_m – буфер фреймов. В блоках JTAG COMM и USB COMM используется аналогичный компонент. Управление компонентом осуществляется через регистры доступные по шине AXI. Формат данных записываемых/считываемых в/из буфера фреймов совпадает с форматом, используемым в JTAG и USB COMM.

49.4.1 Регистры AXI COMM

Перечень регистров приведен в Таблица 49.2.

Таблица 49.2. Перечень программно-доступных регистров AXI COMM

Смещение	Условное обозначение	Название регистра	Исходное состояние	Тип доступа
0	ctr	Управление	0x0	RW
0x0004	sts	Статус	0x0	RO
0x0008	wrd	Запись данных	0x0	RW
0x000C	rdd	Чтение данных	0x0	RO

49.4.1.1 Регистр ctr

Регистр напрямую управляет входами "буфера фреймов".

Формат регистра ctr приведен в следующей таблице:

Таблица 49.3. Поля регистра ctr

Разряды	Название	Описание	Сброс
31:5	-	Резерв	0
4	ust_ack_req	Определяет значение входа ust_ack_req "буфера фреймов". Обычно не используется.	0
3	ust_nack_req	Определяет значение входа ust_nack_req "буфера фреймов". Обычно не используется.	0
2	ust_proterr_req	Определяет значение входа ust_proterr_req "буфера фреймов". Обычно не используется.	0
1	ust_frm_up_flush	Определяет значение входа ust_frm_up_flush "буфера фреймов". Обычно не используется.	0
0	ust_frm_discon	Определяет значение входа ust_frm_discon. Перед началом работы через AXI COMM необходимо установить поле в значение 0. После завершения установить в 1.	1

49.4.1.2 Регистр sts

Регистр отображает текущее состояние выходов "буфера фреймов".

Формат регистра sts приведен в следующей таблице:

Таблица 49.4. Поля регистра sts

Разряды	Название	Описание	Сброс
31:6	-	Резерв	0
5	ust_frm_dn_valid	Признак наличия данных для чтения из буфера фреймов через регистр rdd. Чтение регистра rdd рекомендуется производить только в том случаее, если установлена единица в данном разряде. Попытка чтения из регистра rdd в момент, когда данное поле не установлено, может привести к приостановке выполнения AXI транзакции. Такое чтение завершится только после появления сообщения в буфере фреймов. Но сообщение в буфере фреймов в общем случае может никогда и не появиться.	0
4	ust_frm_connected	Отображает состояние выхода ust_frm_connected буфера фреймов.	0
3	ust_frm_up_full	Отображает состояние заполненности UP FIFO. Признак использовать необязательно. Но следует иметь ввиду, что AXI транзакция записи в регистр wrd будет приостановлена до тех пор, пока буфер фреймов не будет готов к приёму новых данных.	0
2	ust_frm_up_nyet	Отображает состояние выхода ust_frm_up_nyet буфера фреймов. Обычно не используется.	0
1	ust_frm_up_flushed	Отображает состояние выхода ust_frm_up_flushed буфера фреймов. Обычно не используется.	0
0	ust_frm_dn_empty	Признак отсутствия данных в DN FIFO. Отсутствие данных в DN FIFO не означает, что в "буфере фреймов" нет данных, которые можно	0

Разряды	Название	Описание	Сброс
		прочитать, так как сообщения, полученные через DN FIFO, внутри буфера упаковываются дополнительно во фреймы. Также буфер фреймов может самостоятельно генерировать некоторые сообщения без участия DN FIFO (например, отклик на фрейм Initialisation-request(0x000000FF). Для проверки наличия данных в буфере фреймов рекомендуется использовать признак ust_frm_dn_valid.	

49.4.1.3 Регистр wrd

Регистр записи данных сообщений в направлении от отладчика (UP stream). Не все данные записанные в этот регистр попадают в UP FIFO, на некоторые сообщение буфер фреймов откликается сам без участия UP/DN FIFO.

Формат регистра wrd приведен в следующей таблице:

Таблица 49.5. Поля регистра wrd

Разряды	Название	Описание	Сброс
31:0	wrdata	Данные из которых формируются сообщения.	0

49.4.1.4 Регистр rdd

Через регистр считываются сообщения, полученные или сгенерированные буфером фреймов в направлении к отладчику (DOWN stream).

Формат регистра rdd приведен в следующей таблице:

Таблица 49.6. Поля регистра rdd

Разряды	Название	Описание	Сброс
31:0	rddata	Данные DOWN stream.	0

49.5 Управление отладкой

За управление и настройку отладки отвечают следующие регистры:

- service_subs.urb.fuse0.ust_debug_disable – глобальное разрешение отладки (fuse)
- service_subs.urb.ust_dbgen – регистр разрешения отдельных отладочных подсистем
- hsperiph_subs.urb.dbg_ctr – управление подключением USB отладки
- hsperiph_subs.urb.rst – сброс отладки
- hsperiph_subs.ucg1 – включение и настройка частоты отладки
- hsperiph_subs.ucg3 – включение и настройка частоты USB0 PHY

49.6 Особенности работы Virtual Console

Записи в регистр vcon_data должны быть байтовыми (awsize = 0).

Бит int_type (0-й бит) регистра vcon_interrupt_config не имеет сброса. Перед началом работы необходимо проинициализировать этот бит нужным значением.

49.7 Особенности работы Bus Monitor

При включении монитора через set_enabled необходимо, чтобы на все AXI интерфейсы монитора была заведена частота AXI и сняты все ресеты.

49.8 Особенности работы DMA

При первом чтении после включения модуля DMA необходимо дважды отправить команду set_dma_chan.

49.9 Особенности работы USB Ultrasoc Hub

49.9.1 Чтение дескриптора конфигурации Ultrasoc Hub

Если в запросе на чтение дескриптора конфигурации (configuration description) из Ultrasoc Hub поле wLength установить в значение большее, чем поддерживает Ultrasoc Hub, Ultrasoc Hub отвечает пакетом нулевой длины. При работе с Ultrasoc Hub рекомендуется устанавливать значение поля wLength равное 0x19 или же использовать последовательную процедуру считывания:

1. Запросить первые 9 байт и определить по ним реальный размер пакета.
2. Запросить пакет с реальным размером пакета, полученным на шаге один.

50. КОНТРОЛЛЕР ПРЯМОГО ДОСТУПА В ПАМЯТЬ ПОДСИСТЕМЫ SDR (SDRDMA)

50.1 Общая информация

Контроллер прямого доступа в память (SDRDMA) состоит из 8 каналов MEM_CH, каждый из которых обеспечивает обмен 8, 16, 32, 64 или 128 разрядными данными между двумя областями памяти.

50.2 Регистры

В подсистеме SDR работает два 8-канальных контроллера DMA со следующими базовыми адресами регистров:

- 0x0191_8000 – для SDRDMA0
- 0x01C1_8000 – для SDRDMA1

Перечень регистров приведен в Таблица 50.1.

Таблица 50.1. Перечень программно-доступных регистров

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
CSR0	Регистр управления и состояния – CSR 0-го канала	0x0	R/W	0x000
WCX0	32-х разрядный счетчик кол-ва переданных данных WCX 0-го канала	-		0x004
CP0[31:0]	Младшая часть регистра начального 64-х разрядного физического адреса блока параметров для выполнения процедуры самоинициализации 0-го канала. – CP 0-го канала.	-	R/W	0x008
CP0[63:32]	Старшая часть регистра начального 64-х разрядного физического адреса блока параметров для выполнения процедуры самоинициализации 0-го канала. – CP 0-го канала.	-	R/W	0x00C
IR00[31:0]	Младшая часть регистра 0-го 64-х разрядного индекса (физический адрес памяти) – IR0 0-го канала.	-	R/W	0x010
IR00[63:32]	Старшая часть регистра 0-го 64-х разрядного индекса (физический адрес памяти) – IR0 0-го канала.	-	R/W	0x014
IR01[31:0]	Младшая часть регистра 1-го 64-х разрядного индекса (физический адрес памяти) – IR1 0-го канала.	-	R/W	0x018
IR01[63:32]	Старшая часть регистра 1-го 64-х разрядного индекса (физический адрес памяти) – IR1 0-го канала.	-	R/W	0x01C
OR01, OR00	32-х разрядный регистр смещений для 1-го и 0-го индекса 0-го канала Биты 31-16: смещения для 1-го индекса OR1 Биты 15-0: смещения для 0-го индекса OR0	-	R/W	0x020
WCY0,ORY0	32-х разрядный.	-	R/W	0x024

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
	Биты 31-16: счетчик числа строк по Y направлению в 2D режиме. WCY 0-го канала Биты 15-0: смещения между строками по Y направлению в 2D режиме. ORY 0-го канала			
STATUS0	32-х разрядный регистра состояния каналов CSR.	0	R/W	0x028
RUN0	Псевдорегистр RUN управления состоянием 0-го бита регистра CSR 0-го канала	0	W	0x02C
CSR1	Регистр управления и состояния – CSR 1-го канала	0x0	R/W	0x040
WCX1	32-х разрядный счетчик кол-ва переданных данных WCX 1-го канала	-		0x044
CP1[31:0]	Младшая часть регистра начального 64-х разрядного физического адреса блока параметров для выполнения процедуры самоинициализации 1-го канала. – CP 1-го канала.	-	R/W	0x048
CP1[63:32]	Старшая часть регистра начального 64-х разрядного физического адреса блока параметров для выполнения процедуры самоинициализации 1-го канала. – CP 1-го канала.	-	R/W	0x04C
IR10[31:0]	Младшая часть регистра 0-го 64-х разрядного индекса (физический адрес памяти) – IR0 1-го канала.	-	R/W	0x050
IR10[63:32]	Старшая часть регистра 0-го 64-х разрядного индекса (физический адрес памяти) – IR0 1-го канала.	-	R/W	0x054
IR11[31:0]	Младшая часть регистра 1-го 64-х разрядного индекса (физический адрес памяти) – IR1 1-го канала.	-	R/W	0x058
IR11[63:32]	Старшая часть регистра 1-го 64-х разрядного индекса (физический адрес памяти) – IR1 1-го канала.	-	R/W	0x05C
OR11, OR10	32-х разрядный регистр смещений для 1-го и 0-го индекса 1-го канала Биты 31-16: смещения для 1-го индекса OR1 Биты 15-0: смещения для 0-го индекса OR0	-	R/W	0x060
WCY1,ORY1	32-х разрядный. Биты 31-16: счетчик числа строк по Y направлению в 2D режиме. WCY 1-го канала Биты 15-0: смещения между строками по Y направлению в 2D режиме. ORY 1-го канала	-	R/W	0x064
STATUS1	32-х разрядный регистра состояния каналов CSR.	0	R/W	0x068
RUN1	Псевдорегистр RUN управления состоянием 0-го бита регистра CSR 1-го канала	0	W	0x06C
CSR2	Регистр управления и состояния – CSR 2-го канала	0x0	R/W	0x080
WCX2	32-х разрядный счетчик кол-ва переданных данных WCX 2-го канала	-		0x084
CP2[31:0]	Младшая часть регистра начального 64-х разрядного физического адреса блока параметров для выполнения процедуры самоинициализации 2-го канала. – CP 2-го канала.	-	R/W	0x088
CP2[63:32]	Старшая часть регистра начального 64-х разрядного физического адреса блока параметров для выполнения процедуры самоинициализации 2-го канала. – CP 2-го канала.	-	R/W	0x08C

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
IR20[31:0]	Младшая часть регистра 0-го 64-х разрядного индекса (физический адрес памяти) – IR0 2-го канала.	-	R/W	0x090
IR20[63:32]	Старшая часть регистра 0-го 64-х разрядного индекса (физический адрес памяти) – IR0 2-го канала.	-	R/W	0x094
IR21[31:0]	Младшая часть регистра 1-го 64-х разрядного индекса (физический адрес памяти) – IR1 2-го канала.	-	R/W	0x098
IR21[63:32]	Старшая часть регистра 1-го 64-х разрядного индекса (физический адрес памяти) – IR1 2-го канала.	-	R/W	0x09C
OR21, OR20	32-х разрядный регистр смещений для 1-го и 0-го индекса 2-го канала Биты 31-16: смещения для 1-го индекса OR1 Биты 15-0: смещения для 0-го индекса OR0	-	R/W	0x0A0
WCY2,ORY2	32-х разрядный. Биты 31-16: счетчик числа строк по Y направлению в 2D режиме. WCY 2-го канала Биты 15-0: смещения между строками по Y направлению в 2D режиме. ORY 2-го канала	-	R/W	0x0A4
STATUS2	32-х разрядный регистра состояния каналов CSR.	0	R/W	0x0A8
RUN2	Псевдорегистр RUN управления состоянием 0-го бита регистра CSR 2-го канала	0	W	0x0AC
CSR3	Регистр управления и состояния – CSR 3-го канала	0x0	R/W	0x0C0
WCX3	32-х разрядный счетчик кол-ва переданных данных WCX 3-го канала	-		0x0C4
CP3[31:0]	Младшая часть регистра начального 64-х разрядного физического адреса блока параметров для выполнения процедуры самоинициализации 3-го канала. – CP 3-го канала.	-	R/W	0x0C8
CP3[63:32]	Старшая часть регистра начального 64-х разрядного физического адреса блока параметров для выполнения процедуры самоинициализации 3-го канала. – CP 3-го канала.	-	R/W	0x0CC
IR30[31:0]	Младшая часть регистра 0-го 64-х разрядного индекса (физический адрес памяти) – IR0 3-го канала.	-	R/W	0x0D0
IR30[63:32]	Старшая часть регистра 0-го 64-х разрядного индекса (физический адрес памяти) – IR0 3-го канала.	-	R/W	0x0D4
IR31[31:0]	Младшая часть регистра 1-го 64-х разрядного индекса (физический адрес памяти) – IR1 3-го канала.	-	R/W	0x0D8
IR31[63:32]	Старшая часть регистра 1-го 64-х разрядного индекса (физический адрес памяти) – IR1 3-го канала.	-	R/W	0x0DC
OR31, OR30	32-х разрядный регистр смещений для 1-го и 0-го индекса 3-го канала Биты 31-16: смещения для 1-го индекса OR1 Биты 15-0: смещения для 0-го индекса OR0	-	R/W	0x0E0
WCY3,ORY3	32-х разрядный.	-	R/W	0x0E4

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
	Биты 31-16: счетчик числа строк по Y направлению в 2D режиме. WCY 3-го канала Биты 15-0: смещения между строками по Y направлению в 2D режиме. ORY 3-го канала			
STATUS3	32-х разрядный регистра состояния каналов CSR.	0	R/W	0x0E8
RUN3	Псевдорегистр RUN управления состоянием 0-го бита регистра CSR 3-го канала	0	W	0x0EC
CSR4	Регистр управления и состояния – CSR 4-го канала	0x0	R/W	0x100
WCX4	32-х разрядный счетчик кол-ва переданных данных WCX 4-го канала	-		0x104
CP4[31:0]	Младшая часть регистра начального 64-х разрядного физического адреса блока параметров для выполнения процедуры самоинициализации 4-го канала. – CP 4-го канала.	-	R/W	0x108
CP4[63:32]	Старшая часть регистра начального 64-х разрядного физического адреса блока параметров для выполнения процедуры самоинициализации 4-го канала. – CP 4-го канала.	-	R/W	0x10C
IR40[31:0]	Младшая часть регистра 0-го 64-х разрядного индекса (физический адрес памяти) – IR0 4-го канала.	-	R/W	0x110
IR40[63:32]	Старшая часть регистра 0-го 64-х разрядного индекса (физический адрес памяти) – IR0 4-го канала.	-	R/W	0x114
IR41[31:0]	Младшая часть регистра 1-го 64-х разрядного индекса (физический адрес памяти) – IR1 4-го канала.	-	R/W	0x118
IR41[63:32]	Старшая часть регистра 1-го 64-х разрядного индекса (физический адрес памяти) – IR1 4-го канала.	-	R/W	0x11C
OR41, OR40	32-х разрядный регистр смещений для 1-го и 0-го индекса 4-го канала Биты 31-16: смещения для 1-го индекса OR1 Биты 15-0: смещения для 0-го индекса OR0	-	R/W	0x120
WCY4,ORY4	32-х разрядный. Биты 31-16: счетчик числа строк по Y направлению в 2D режиме. WCY 4-го канала Биты 15-0: смещения между строками по Y направлению в 2D режиме. ORY 4-го канала	-	R/W	0x124
STATUS4	32-х разрядный регистра состояния каналов CSR.	0	R/W	0x128
RUN4	Псевдорегистр RUN управления состоянием 0-го бита регистра CSR 4-го канала	0	W	0x12C
CSR5	Регистр управления и состояния – CSR 5-го канала	0x0	R/W	0x140
WCX5	32-х разрядный счетчик кол-ва переданных данных WCX 5-го канала	-		0x144
CP5[31:0]	Младшая часть регистра начального 64-х разрядного физического адреса блока параметров для выполнения процедуры самоинициализации 5-го канала. – CP 5-го канала.	-	R/W	0x148
CP5[63:32]	Старшая часть регистра начального 64-х разрядного физического адреса блока параметров для выполнения процедуры самоинициализации 5-го канала. – CP 5-го канала.	-	R/W	0x14C

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
IR50[31:0]	Младшая часть регистра 0-го 64-х разрядного индекса (физический адрес памяти) – IR0 5-го канала.	-	R/W	0x150
IR50[63:32]	Старшая часть регистра 0-го 64-х разрядного индекса (физический адрес памяти) – IR0 5-го канала.	-	R/W	0x154
IR51[31:0]	Младшая часть регистра 1-го 64-х разрядного индекса (физический адрес памяти) – IR1 5-го канала.	-	R/W	0x158
IR51[63:32]	Старшая часть регистра 1-го 64-х разрядного индекса (физический адрес памяти) – IR1 5-го канала.	-	R/W	0x15C
OR51, OR50	32-х разрядный регистр смещений для 1-го и 0-го индекса 5-го канала Биты 31-16: смещения для 1-го индекса OR1 Биты 15-0: смещения для 0-го индекса OR0	-	R/W	0x160
WCY5,ORY5	32-х разрядный. Биты 31-16: счетчик числа строк по Y направлению в 2D режиме. WCY 5-го канала Биты 15-0: смещения между строками по Y направлению в 2D режиме. ORY 5-го канала	-	R/W	0x164
STATUS5	32-х разрядный регистра состояния каналов CSR.	0	R/W	0x168
RUN5	Псевдорегистр RUN управления состоянием 0-го бита регистра CSR 5-го канала	0	W	0x16C
CSR6	Регистр управления и состояния – CSR 6-го канала	0x0	R/W	0x180
WCX6	32-х разрядный счетчик кол-ва переданных данных WCX 6-го канала	-		0x184
CP6[31:0]	Младшая часть регистра начального 64-х разрядного физического адреса блока параметров для выполнения процедуры самоинициализации 6-го канала. – CP 6-го канала.	-	R/W	0x188
CP6[63:32]	Старшая часть регистра начального 64-х разрядного физического адреса блока параметров для выполнения процедуры самоинициализации 6-го канала. – CP 6-го канала.	-	R/W	0x18C
IR60[31:0]	Младшая часть регистра 0-го 64-х разрядного индекса (физический адрес памяти) – IR0 6-го канала.	-	R/W	0x190
IR60[63:32]	Старшая часть регистра 0-го 64-х разрядного индекса (физический адрес памяти) – IR0 6-го канала.	-	R/W	0x194
IR61[31:0]	Младшая часть регистра 1-го 64-х разрядного индекса (физический адрес памяти) – IR1 6-го канала.	-	R/W	0x198
IR61[63:32]	Старшая часть регистра 1-го 64-х разрядного индекса (физический адрес памяти) – IR1 6-го канала.	-	R/W	0x19C
OR61, OR60	32-х разрядный регистр смещений для 1-го и 0-го индекса 6-го канала Биты 31-16: смещения для 1-го индекса OR1 Биты 15-0: смещения для 0-го индекса OR0	-	R/W	0x1A0
WCY6,ORY6	32-х разрядный.	-	R/W	0x1A4

Условное обозначение	Название регистра	Исходное состояние	Тип доступа	Смещение
	Биты 31-16: счетчик числа строк по Y направлению в 2D режиме. WCY 6-го канала Биты 15-0: смещения между строками по Y направлению в 2D режиме. ORY 6-го канала			
STATUS6	32-х разрядный регистра состояния каналов CSR.	0	R/W	0x1A8
RUN6	Псевдорегистр RUN управления состоянием 0-го бита регистра CSR 6-го канала	0	W	0x1AC
CSR7	Регистр управления и состояния – CSR 7-го канала	0x0	R/W	0x1C0
WCX7	32-х разрядный счетчик кол-ва переданных данных WCX 7-го канала	-		0x1C4
CP7[31:0]	Младшая часть регистра начального 64-х разрядного физического адреса блока параметров для выполнения процедуры самоинициализации 7-го канала. – CP 7-го канала.	-	R/W	0x1C8
CP7[63:32]	Старшая часть регистра начального 64-х разрядного физического адреса блока параметров для выполнения процедуры самоинициализации 7-го канала. – CP 7-го канала.	-	R/W	0x1CC
IR70[31:0]	Младшая часть регистра 0-го 64-х разрядного индекса (физический адрес памяти) – IR0 7-го канала.	-	R/W	0x1D0
IR70[63:32]	Старшая часть регистра 0-го 64-х разрядного индекса (физический адрес памяти) – IR0 7-го канала.	-	R/W	0x1D4
IR71[31:0]	Младшая часть регистра 1-го 64-х разрядного индекса (физический адрес памяти) – IR1 7-го канала.	-	R/W	0x1D8
IR71[63:32]	Старшая часть регистра 1-го 64-х разрядного индекса (физический адрес памяти) – IR1 7-го канала.	-	R/W	0x1DC
OR71, OR70	32-х разрядный регистр смещений для 1-го и 0-го индекса 7-го канала Биты 31-16: смещения для 1-го индекса OR1 Биты 15-0: смещения для 0-го индекса OR0	-	R/W	0x1E0
WCY7,ORY7	32-х разрядный. Биты 31-16: счетчик числа строк по Y направлению в 2D режиме. WCY 7-го канала Биты 15-0: смещения между строками по Y направлению в 2D режиме. ORY 7-го канала	-	R/W	0x1E4
STATUS7	32-х разрядный регистра состояния каналов CSR.	0	R/W	0x1E8
RUN7	Псевдорегистр RUN управления состоянием 0-го бита регистра CSR 7-го канала	0	W	0x1EC

50.2.1 Регистр CSR

Регистр CSR.

Таблица 50.2. Формат регистра CSR

Номер бита	Условное обозначение	Назначение
31:15	-	Резерв

Номер бита	Условное обозначение	Назначение
14:13	MASK	Маска внешнего запроса прямого доступа nDMAR: 0 – запрос запрещен; 1 – запрос разрешен, по сигналу DMAR передается (WN+1) слов 2 – запрос разрешен, по сигналу DMAR передается (WCX+1) слов 3 – запрос разрешен, по сигналу DMAR передается (WCX+1)*(WCY+1) слов, если установлен двумерный режим передачи данных (2D), иначе передается (WCX+1) слов Если поле равно нулю, то канал работает только под управлением бита RUN. Если поле не равно нулю, то для инициализации канала необходимо также наличие запроса nDMAR (низкий уровень).
12	CHEN	Признак разрешения самоинициализации (выполнения цепочки DMA передач)
11	IM	Разрешение установки признака окончания передачи блока данных: 0 – установки признака запрещено; 1 – установки признака разрешено.
10:8	SIZE	Размерность передаваемых данных 000 – 8 разрядов 001 – 16 разрядов 010 – 32 разряда 011 – 64 разряда 100 – 128 разрядов Остальные коды резервные (по умолчанию 128 разрядов)
7:4	WN	Число слов данных (пачка), которое передается каналом за одно предоставление прямого доступа: 0 – 1 слово, F – 16 слов. Посредством этого параметра можно плавно изменять приоритет каналов DMA относительно друг друга и относительно других устройств.
3	2D	Режим модификации адреса регистра IR1: 0 – одномерный режим; 1 – двухмерный режим.
2	REV	Режим модификации адреса регистра IR0: 0 – линейный режим; 1 – режим с обратным переносом.
1	DIR	Направление обмена данными: 0 – память по IR0 => память по IR1; 1 – память по IR1 => память по IR0.
0	RUN	Состояние работы канала DMA: 0 – состояние останова; 1 – состояние обмена данными.

50.2.2 Регистр STATUS

Регистр STATUS.

Таблица 50.3. Формат регистра STATUS

Номер бита	Условное обозначение	Назначение
31:12	-	Резерв
11:10	BRESP[1:0]	Признак ошибки AXI коммутатора BRESP. Выводится на внешний вывод прерывания IRQ, соответствующему данному каналу, по «или» с битами DONE, END и RRESP(1). Сбрасывается в «0» записью «1» в этот разряд.
9:8	RRESP[1:0]	Признак ошибки AXI коммутатора RRESP. Выводится на внешний вывод прерывания IRQ, соответствующему данному каналу, по «или» с битами DONE, END и BRESP(1). Сбрасывается в «0» записью «1» в этот разряд.
7:2	-	Резерв
1	DONE	Признак завершения передачи цепочки блоков данных. Аппаратно устанавливается в 1 после завершения передачи цепочки блоков данных при CHEN=0, при этом бит RUN сбрасывается. Выводится на внешний вывод прерывания IRQ, соответствующему данному каналу, по «или» с битами END, RRESP(1) и BRESP(1). Сбрасывается в «0» записью «1» в этот разряд.
0	END	Признак окончания передачи блока данных. Аппаратно устанавливается в 1 после завершения передачи блока данных при IM=1. Выводится на внешний вывод прерывания IRQ, соответствующему данному каналу, по «или» с битами DONE, RRESP(1) и BRESP(1). Сбрасывается в «0» записью «1» в этот разряд.

50.3 Функционирование

Для управления работой каждого канала MEM_CH имеются следующие регистры:

- регистр управления – CSR;
- регистр состояния – STATUS;
- регистры счетчиков количества переданных данных WCX, WCY;
- регистры индекса (физический адрес памяти) – IR0, IR1;
- регистры смещения – OR0, OR1, ORY;
- регистр начального физического адреса блока параметров для выполнения процедуры самоинициализации - CP;
- псевдорегистр RUN управления состоянием 0-го бита регистра CSR.

Исходное состояние регистров CSR и STATUS – нули.

Исходное состояние остальных регистров не определено.

50.3.1 Организация обмена данными.

Бит RUN регистра CSR равный 1 является запросом на обмен “память-память” под управлением канала ДМА. Бит RUN регистра CSR автоматически сбрасывается в 0 и биты END и DONE регистра STATUS устанавливаются в 1 при выдаче в AXI коммутатор последних заказанных данных. Все разряды регистра CSR доступны по записи и чтению.

Состоянием разряда 0 (бит RUN) регистра CSR можно также управлять, используя запись по адресу псевдорегистра RUN. При этом остальные разряды этого регистра не изменяются. Эта процедура может быть использована как для временной приостановки канала DMA, так и для старта канала с заранее подготовленными данными или с точки останова.

Регистр состояния каналов STATUS содержит признаки прерываний от канала.

Все значимые разряды регистра доступны по чтению. Резервные разряды читаются нулем.

Флаги прерываний сбрасываются записью “1” в соответствующий разряд. Запись “0” в соответствующий разряд сохраняет значение.

64-разрядные регистры индекса IR0, IR1 содержат начальные физические адреса источника и приемника данных (или, наоборот, в зависимости от содержимого разряда DIR регистра CSR). В зависимости от содержимого поля SIZE адреса в этих регистрах должны быть выровнены по границе 8, 16, 32, 64 или 128-разрядного слова. OR0 - смещение (приращение) адреса для индексного регистра IR0 после передачи каждого слова данных. OR1 - смещение (приращение) адреса для индексного регистра IR1 после передачи каждого слова данных.

Модификация индексных регистров IR0, IR1 при помощи смещений OR0, OR1 может обеспечиваться в режиме с прямым переносом.

В режиме модификации индексного регистра с прямым переносом смещение, задаваемое полями OR0, OR1, рассматривается как число со знаком в диапазоне –32768 до +32767 слов данных (128, 64, 32, 16 или 8-разрядных). Алгоритм модификации адреса с прямым переносом:

for (x = 0; x < WCX; x++) { пересылка по адресу IR0;

модификация адреса для 128-ми разрядного обмена:

IR0[63:0] = IR0[63:0] + {{44{OR0[15]}},OR0[15:0],0000};

модификация адреса для 64-х разрядного обмена:

IR0[63:0] = IR0[63:0] + {{45{OR0[15]}},OR0[15:0],000};

модификация адреса для 32-х разрядного обмена:

IR0[63:0] = IR0[63:0] + {{46{OR0[15]}},OR0[15:0],00};

модификация адреса для 16-ти разрядного обмена:

IR0[63:0] = IR0[63:0] + {{47{OR0[15]}},OR0[15:0],0};

модификация адреса для 8-ми разрядного обмена:

IR0[63:0] = IR0[63:0] + {{48{OR0[15]}},OR0[15:0]};

пересылка по адресу IR1;

модификация адреса для 128-ми разрядного обмена:

IR1[63:0] = IR1[63:0] + {{43{OR1[15]}},OR1[15:0],0000};

модификация адреса для 64-х разрядного обмена:

IR1[63:0] = IR1[63:0] + {{44{OR1[15]}},OR1[15:0],000};

модификация адреса для 32-х разрядного обмена:

IR1[63:0] = IR1[63:0] + {{46{OR1[15]}},OR1[15:0],00};

модификация адреса для 16-ти разрядного обмена:

IR1[63:0] = IR1[63:0] + {{47{OR1[15]}},OR1[15:0],0};

модификация адреса для 8-ми разрядного обмена:

IR1[63:0] = IR1[63:0] + {{48{OR1[15]}},OR1[15:0]};

}

Режим модификации индексного регистра с обратным переносом используется только для IR0.

В режиме модификации индексного регистра с обратным переносом смещение, задаваемое полем OR0, имеет диапазон от 0 до +65535. Модификация адреса в этом случае выполняется с распространением переноса в обратном направлении – от старших разрядов к младшим. Операция модификации адреса с обратным переносом эквивалентна последовательному выполнению следующих процедур:

- 16-разрядное смещение OR0 дополняется до 64 разрядов:
- для 128-разрядного обмена – со стороны младших четырьмя нулями, а со стороны старших 44-мя нулями.
- для 64-разрядного обмена – со стороны младших тремя нулями, а со стороны старших 45-ю нулями.
- для 32-разрядного обмена – со стороны младших двумя нулями, а со стороны старших 46-ю нулями.
- для 16-разрядного обмена – со стороны младших одним нулем, а со стороны старших 47-ю нулями.
- для 8-разрядного обмена – со стороны старших 48-ю нулями.
- изменение на обратный порядок следования разрядов в регистрах адреса и смещения. При этом старший бит становится младшим;
- модификация адреса посредством операции сложения с прямым переносом;
- восстановление первоначального порядка следования разрядов регистра адреса.

Модификацию адреса с обратным переносом можно описать при помощи следующих выражений:

IR0 [0:63] = IR0[0:63] + {0000, OR0[0:15], 44 нуля } – для 128-разрядного обмена;

IR0 [0:63] = IR0[0:63] + { 000, OR0[0:15], 45 нулей } – для 64-разрядного обмена.

IR0 [0:63] = IR0[0:63] + { 00, OR0[0:15], 46 нулей } – для 32-разрядного обмена.

IR0 [0:63] = IR0[0:63] + { 0, OR0[0:15], 47 нулей } – для 16-разрядного обмена.

IR0 [0:63] = IR0[0:63] + { OR0[0:15], 48 нулей } – для 8-разрядного обмена.

Канал MEM_CH обеспечивают передачу двумерных массивов (матриц W[m;n]). Данный режим может быть реализован только для IR1. При этом память адресуется в двухмерном режиме. При двухмерном режиме адресации поле WCX регистра CSR содержит число слов в строке строк = WCX + 1 (X направление), а поле WCY содержит число строк = WCY + 1 (Y направление). ORY - смещение (приращение) адреса памяти по направлению Y. Смещение, задаваемое полем ORY, рассматривается как число со знаком в диапазоне – 32768 до +32767 слов данных (128, 64, 32, 16 или 8-разрядных). Счетчик WCY и смещение ORY используется только при двухмерной адресации.

Пересылка каждого слова данных осуществляется по индексному регистру IR1 с его последующей инкрементацией на величину, соответствующую содержимому поля OR1 (X направление) или поля ORY (переход на новую строку Y направление). Двухмерная адресация выполняется следующим образом:

Содержимое счетчика WCX сохраняется в буферном регистре;

1 цикл. Индексный регистр внешней памяти модифицируется с использованием смещения OR1. Счетчик WCX декрементируется. Если он равен 0, то переход ко второму циклу.

2 цикл. Состояние счетчика WCX восстанавливается из буферного регистра. Индексный регистр внешней памяти модифицируется с использованием смещения ORY. Счетчик WCY декрементируется. Если он не равен 0, то переход к первому циклу. Если он равен 0, то работа канала завершается.

Функционально двумерная адресация эквивалентна следующему двойному циклу:

```
for ( y = 0; y <= WCY; y++ ) {
  for ( x = 0; x < WCX; x++ ) { пересылка по адресу IR1
для 128-ми разрядного обмена : IR1[63:0] = IR1[63:0] + {{44{OR1[15]}},OR1[15:0],0000};
для  64-х  разрядного обмена : IR1[63:0] = IR1[63:0] + {{45{OR1[15]}},OR1[15:0],000};
для  32-х  разрядного обмена : IR1[63:0] = IR1[63:0] + {{46{OR1[15]}},OR1[15:0],00}
для  16-ти разрядного обмена : IR1[63:0] = IR1[63:0] + {{47{OR1[15]}},OR1[15:0],0};
для   8-ми разрядного обмена : IR1[63:0] = IR1[63:0] + {{48{OR1[15]}},OR1[15:0]}
```

};

пересылка по адресу IR1

для 128-ми разрядного обмена : IR1[63:0] = IR1[63:0] + {{44{ORY[15]}},ORY[15:0],0000};

для 64-х разрядного обмена : IR1[63:0] = IR1[63:0] + {{45{ORY[15]}},ORY[15:0],000};

для 32-х разрядного обмена : IR1[63:0] = IR1[63:0] + {{46{ORY[15]}},ORY[15:0],00};

для 16-ти разрядного обмена : IR1[63:0] = IR1[63:0] + {{47{ORY[15]}},ORY[15:0],0};

для 8-ми разрядного обмена : IR1[63:0] = IR1[63:0] + {{48{ORY[15]}},ORY[15:0]};

};

//общее кол-во пересылок (WCX+1)*(WCY+1)

Все каналы DMA могут выполнять процедуру самоинициализации (выполнение цепочки передач DMA).

Для выполнения самоинициализации в каналах DMA имеется 64-разрядный регистр CP, в котором хранится физический начальный адрес блока параметров очередного DMA обмена. Младшие три разряда регистра CP при формировании адреса самоинициализации заменяются нулями (адреса блоков информации для самоинициализации должны быть выровнены по границе 64-разрядного слова). Параметры для самоинициализации каналов DMA MEM_CH размещаются в памяти в пяти последовательных 64-разрядных словах, следующим образом (в порядке возрастания адресов):

63_____0

{ IR0(63:0) }; // адрес = {CP[63:3],000}

{ IR1(63:0), }; // адрес = {CP[63:3],000} + 1000

{{WCY(15:0),ORY(15:0)},{ OR1(15:0),OR0(15:0) }}; // адрес = {CP[63:3],000} + 10000

{ CP(63:0) }; // адрес = {CP[63:3],000} + 11000

{ {WCX(31:0),CSR(31:0)} }. // адрес = {CP[63:3],000} + 100000

Младший (нулевой разряд) регистра CP используется для старта режима самоинициализации одновременно с записью в младшую часть регистра CP по AHB младшей части начального адреса параметров для самоинициализаци. Если он равен 1, то происходит автоматический старт самоинициализации по загружаемому в регистр CP адресу ({CP[63:3],000}). Параметры при самоинициализации аппаратно загружаются в 64-разрядном формате в соответствующие регистры канала DMA. Процедура загрузки параметров встроена в DMA обмены с коммутатора AXI. Блок параметров может размещаться в любой доступной памяти.

Если необходимо продолжить цепочку команд, то необходимо указать CHEN=1. В режиме самоинициализации при записи параметров в регистр CSR биты END и DONE недоступны.

Для запуска работы канала DMA в режиме с самоинициализацией необходимо в регистр CP записать адрес первого блока параметров DMA передачи. При этом 0 разряд записываемых данных должен содержать 1 (признак пуска самоинициализации). В результате этого, соответствующий канал загрузит в свои регистры параметры DMA передачи и начнет обмен данными, если бит RUN регистра CSR равен 1. После окончания передачи блока данных если бит IM = 1, то бит END в регистре CSR устанавливается в единичное состояние и выдается прерывание. По окончании передачи блока данных также проверяется состояние бита CHEN. Если он равен 1, то будет загружен следующий блок параметров DMA передачи и т.д. В противном случае цепочка DMA обменов закончится и в регистре CSR бит DONE установится в единичное состояние и выдается прерывание. Бит RUN регистра CSR установится 0.

При необходимости каналы DMA могут инициализироваться программно через протокол AHB. Для этого CPU должен загрузить все необходимые регистры индекса и смещения, а затем регистр CSR. При загрузке регистра CSR бит RUN необходимо установить в единичное состояние. Младший разряд регистра CP должен быть равен 0, а сам регистр должен содержать адрес параметров для следующего звена цепочки при CHEN=1. Следует отметить, что бит RUN регистра CSR может быть использован для приостановки выполнения каналом DMA цепочки. Для этого в любой момент времени в него необходимо записать 0. Для продолжения работы соответственно в бит RUN регистра CSR необходимо записать 1. Бит RUN регистра CSR может быть использован также для приостановки выполнения цепочки, если при загрузке очередных параметров он будет равен 0. Для продолжения выполнения цепочки в бит RUN регистра CSR необходимо записать 1. Для удобства организации обмена только с битом RUN регистра CSR имеется специальный псевдорегистр RUN.

50.3.2 Внешние сигналы запроса прямого доступа nDMAR.

DMA имеет 8 (по кол-ву каналов) внешних сигналов запроса прямого доступа nDMAR[7:0]. Эти сигналы поступают на каналы DMA MEM_CH следующим образом:

nDMAR[0] – на каналы MEM_CH0;

nDMAR[1] – на каналы MEM_CH1;

nDMAR[2] – на каналы MEM_CH2;

nDMAR[3] – на каналы MEM_CH3;

nDMAR[4] – на каналы MEM_CH4;

nDMAR[5] – на каналы MEM_CH5;

nDMAR[6] – на каналы MEM_CH6;

nDMAR[7] – на каналы MEM_CH7;

Для настройки работы канала DMA MEM_CH по внешним запросам необходимо в регистре CSR установить: MASK≠0, RUN=1. Внешнее устройство необходимо активизировать на формирование сигналов nDMAR, только после настройки соответствующего канала DMA MEM_CH.

По каждому переходу сигнала nDMAR из 1 в 0 канал DMA MEM_CH выполняет процедуру передачи одного пакета слов данных размером в соответствии с полем WN регистра CSR_MEM_CH. Очередной сигнал запроса прямого доступа запрещается подавать до окончания процесса передачи предыдущего заказа слов данных. Реализовано три режима передачи данных по внешнему запросу:

При MASK =1 по сигналу DMAR передается (WN+1) слов данных.

При MASK =2 по сигналу DMAR передается (WCX+1) слов данных.

При MASK =3 по сигналу DMAR передается (WCX+1)*(WCY+1) слов, если установлен признак двумерного режима передачи данных (2D), иначе передается (WCX+1) слов

Необходимо иметь в виду, что факт перехода сигнала nDMAR из 1 в 0 запоминается в DMA только при RUN=1, MASK≠0. При выполнении любой операции записи в регистр CSR, сбрасывается запомненный в DMA факт перехода сигнала nDMAR из 1 в 0, если он не был принят к исполнению к этому моменту.

50.3.3 Прерывания DMA

Бит DONE регистра STATUS устанавливаются в 1 при выдаче в AXI коммутатор последних данных всей заказанной цепочки.

Бит END регистра STATUS устанавливаются в 1 при выдаче в AXI коммутатор последних данных звена заказанной цепочки если бит IM регистра CSR равен 1.

В случае возникновения ошибок передачи данных RRESP, BRESP (равными 10 или 11), канал останавливается, коды ошибок прописываются в соответствующие поля регистра STATUS.

Канал DMA формирует прерывания от каналов при единичном состоянии битов DONE, END, по кодам ошибок передачи данных RRESP, BRESP равным 10 или 11 регистра STATUS соответствующего канала собранными по “или”.

Обнуление битов END, DONE и полей RRESP, BRESP (и снятие соответствующего прерывания) выполняется посредством записи “1” в соответствующие биты регистра STATUS.

51. ОПИСАНИЕ ВНЕШНИХ ВЫВОДОВ

Параметры корпуса микросхемы:

- тип корпуса: HFCBGA;
- размер корпуса (габариты): 23 x 23 мм;
- шаг по выводам: 0,5 мм;
- матрица выводов: 44 x 44;
- общее число выводов: 1936.

Перечень сигналов микросхемы по группам, приведен в Таблица 51.1.

Таблица 51.1. Перечень сигналов по группам

Назначение	Число выводов
Порты внешней памяти DDRMC0, DDRMC1	184
Порт внешней памяти NANDMPORT	25
Порт внешней памяти QSPI0, QSPI1	18
Порты внешней памяти SDMMC0, SDMMC1	28
Порты видео ввода/вывода	45
Порты видео ввода/вывода MIPI CSI, DSI	34
Порты MFBSP0, MFBSP1	20
Выводы GPIO, периферийные устройства LSPERIPH0 и LSPERIPH1	64
Порт EMAC0, EMAC1	28
Порт USB0, USB1	24
Интерфейс коррелятора GNSS	14
Универсальные CMOS выводы SDR для подключения GPIO/SPI/ADC	32
LVDS выводы для подключения к АЦП/ЦАП	74
Служебные LVDS IO	16
CMOS выводы для подключения к ЦАП	17
Дополнительные выводы для подключения ЦАП/АЦП	5
Выводы PCIe0, PCIe1/JESD204B	60
Служебные выводы	21
Итого	707

Все неиспользуемые выводы типа «I», «IO» необходимо подключить к земле, если в этих таблицах не указано иное требование, или подтяжка этих выводов не обеспечивается самой микросхемой.

При сопряжении данной микросхемы с внешними устройствами, например, памятью, в зависимости от параметров платы, необходимо устанавливать схемы последовательного или параллельного согласования. Необходимость их установки определяет разработчик аппаратуры самостоятельно.

51.1 Сигнальные выводы

Примечание. Для мультиплексированных с GPIO выводов в приведенных ниже таблицах исходное состояние указано в столбце «Функция 0».

Примечание. В поле Drive, приведенных ниже таблиц, указано значение выходного тока для контактной площадки в мА по сбросу.

Примечание. В поле Pull up/down, приведенных ниже таблиц, указано наличие внутреннего подтягивающего резистора контактной площадки. Номиналы резисторов указаны в разделе 34.

Примечание. Поле «Тип вывода» может иметь следующие значения:

I - вход

O - выход (только активное состояние)

I/O - вход/выход

NC - not connected

Примечание. Для дифференциальных сигналов используются различные наименования для положительного и отрицательного выводов сигнала. Используются следующие пары наименований:

- signal – положительный; signalX – отрицательный;
- signal_P – положительный; signal_M – отрицательный;
- signalPx – положительный; signalNx – отрицательный;

51.1.1 Выводы DDRMC0

Таблица 51.2. Перечень сигнальных выводов. DDRMC0

Название вывода	Вывод на корпусе	DDR3	DDR4	LPDDR3	LPDDR4	Назначение	Домен питания
DDR0_MEM RESET_L	E39	RESET_N	RESET_N	NC	RESET_N	Сигнал сброса	DDR0_VDDQ
DDR0_ALERT_N	E40	NC	ALERT_N	NC	NC	Сигнализация ошибки от DRAM	
DDR0_D0	AA44	DQ0	DQ0	DQA0	DQA0	В режиме DDR3 и DDR4 вывод используется как шина данных DQ[7:0] В режиме LPDDR3 и LPDDR4 вывод используется как шина данных DQA[7:0]	
DDR0_D1	Y44	DQ1	DQ1	DQA1	DQA1		
DDR0_D2	AA43	DQ2	DQ2	DQA2	DQA2		
DDR0_D3	Y42	DQ3	DQ3	DQA3	DQA3		
DDR0_D4	T43	DQ4	DQ4	DQA4	DQA4		
DDR0_D5	T42	DQ5	DQ5	DQA5	DQA5		
DDR0_D6	R44	DQ6	DQ6	DQA6	DQA6		
DDR0_D7	T44	DQ7	DQ7	DQA7	DQA7		
DDR0_D8	W44	DM0 x4: DQS_T9	DM0/ DBI0 x4: DQS_T9	DMA0	DMA0/ DBIA0	В режиме DDR3 вывод используется как маска записи данных (DM0), а при использовании микросхем с шириной шины данных x4 как строб записи данных (DQS_T9) В режиме DDR4 вывод используется как маска записи данных (DM0) или признак инверсии шины данных (DBI0), а при использовании микросхем с шириной шины данных x4 как строб записи данных (DQS_T9) В режиме LPDDR3 вывод используется как маска записи данных (DMA0) В режиме LPDDR4 вывод используется как маска записи данных (DMA0) или признак инверсии шины данных (DBIA0)	
DDR0_D9	V44	DQS_T0	DQS_T0	DQSA_T0	DQSA_T0	В режиме DDR3 и DDR4 вывод используется как строб записи данных DQS_*0 В режиме LPDDR3 и LPDDR4 вывод используется как строб записи данных DQSA_*0	
DDR0_D10	U44	DQS_C0	DQS_C0	DQSA_C0	DQSA_C0		

Название вывода	Вывод на корпусе	DDR3	DDR4	LPDDR3	LPDDR4	Назначение	Домен питания
DDR0_D11	W43	NC x4: DQS_C9	NC x4: DQS_C9	NC	NC	Вывод используется как строб записи данных (DQS_C9) только в режимах DDR3 и DDR4 и только при использовании микросхем с шириной шины данных x4	
DDR0_D12	N42	DQ8	DQ8	DQA8	DQA8	В режиме DDR3 и DDR4 вывод используется как шина данных DQ[15:8] В режиме LPDDR3 и LPDDR4 вывод используется как шина данных DQA[15:8]	
DDR0_D13	N44	DQ9	DQ9	DQA9	DQA9		
DDR0_D14	P44	DQ10	DQ10	DQA10	DQA10		
DDR0_D15	P43	DQ11	DQ11	DQA11	DQA11		
DDR0_D16	H44	DQ12	DQ12	DQA12	DQA12		
DDR0_D17	J44	DQ13	DQ13	DQA13	DQA13		
DDR0_D18	J42	DQ14	DQ14	DQA14	DQA14		
DDR0_D19	J43	DQ15	DQ15	DQA15	DQA15		
DDR0_D20	M44	DM1 x4: DQS_T10	DM1/ DBI1 x4: DQS_T10	DMA1	DMA1/ DBIA1	В режиме DDR3 вывод используется как маска записи данных (DM1), а при использовании микросхем с шириной шины данных x4 как строб записи данных (DQS_T10) В режиме DDR4 вывод используется как маска записи данных (DM1) или признак инверсии шины данных (DBI1), а при использовании микросхем с шириной шины данных x4 как строб записи данных (DQS_T10) В режиме LPDDR3 вывод используется как маска записи данных (DMA1) В режиме LPDDR4 вывод используется как маска записи данных (DMA1) или признак инверсии шины данных (DBIA1)	
DDR0_D21	L44	DQS_T1	DQS_T1	DQSA_T1	DQSA_T1	В режиме DDR3 и DDR4 вывод используется как строб записи данных DQS_*1 В режиме LPDDR3 и LPDDR4 вывод используется как строб записи данных DQSA_*1	
DDR0_D22	K44	DQS_C1	DQS_C1	DQSA_C1	DQSA_C1		
DDR0_D23	M43	NC x4: DQS_C10	NC x4: DQS_C10	NC	NC	Вывод используется как строб записи данных (DQS_C10) только в режимах DDR3 и DDR4 и только при использовании микросхем с шириной шины данных x4	
DDR0_D24	B44	DQ16	DQ16	DQB0	DQB0	В режиме DDR3 и DDR4 вывод используется как шина данных DQ[23:16]	
DDR0_D25	C44	DQ17	DQ17	DQB1	DQB1		
DDR0_D26	F44	DQ18	DQ18	DQB2	DQB2		

Название вывода	Вывод на корпусе	DDR3	DDR4	LPDDR3	LPDDR4	Назначение	Домен питания
DDR0_D27	G44	DQ19	DQ19	DQB3	DQB3	В режиме LPDDR3 и LPDDR4 вывод используется как шина данных DQB[7:0]	
DDR0_D28	G43	DQ20	DQ20	DQB4	DQB4		
DDR0_D29	F42	DQ21	DQ21	DQB5	DQB5		
DDR0_D30	C43	DQ22	DQ22	DQB6	DQB6		
DDR0_D31	A44	DQ23	DQ23	DQB7	DQB7		
DDR0_D32	D44	DM2 x4: DQS_T11	DM2/ DBI2 x4: DQS_T11	DMB0	DMB0/ DBIB0	В режиме DDR3 вывод используется как маска записи данных (DM2), а при использовании микросхем с шириной шины данных x4 как строб записи данных (DQS_T11) В режиме DDR4 вывод используется как маска записи данных (DM2) или признак инверсии шины данных (DBI2), а при использовании микросхем с шириной шины данных x4 как строб записи данных (DQS_T11) В режиме LPDDR3 вывод используется как маска записи данных (DMB0) В режиме LPDDR4 вывод используется как маска записи данных (DMB0) или признак инверсии шины данных (DBIB0)	
DDR0_D33	E44	DQS_T2	DQS_T2	DQSB_T0	DQSB_T0	В режиме DDR3 и DDR4 вывод используется как строб записи данных DQS_*2 В режиме LPDDR3 и LPDDR4 вывод используется как строб записи данных DQSA_*2	
DDR0_D34	E43	DQS_C2	DQS_C2	DQSB_C0	DQSB_C0		
DDR0_D35	D42	NC x4: DQS_C11	NC x4: DQS_C11	NC	NC	Вывод используется как строб записи данных (DQS_C11) только в режимах DDR3 и DDR4 и только при использовании микросхем с шириной шины данных x4	
DDR0_D36	C39	DQ24	DQ24	DQB8	DQB8	В режиме DDR3 и DDR4 вывод используется как шина данных DQ[31:24] В режиме LPDDR3 и LPDDR4 вывод используется как шина данных DQB[15:7]	
DDR0_D37	A39	DQ25	DQ25	DQB9	DQB9		
DDR0_D38	A42	DQ26	DQ26	DQB10	DQB10		
DDR0_D39	A43	DQ27	DQ27	DQB11	DQB11		
DDR0_D40	B43	DQ28	DQ28	DQB12	DQB12		
DDR0_D41	C42	DQ29	DQ29	DQB13	DQB13		
DDR0_D42	A38	DQ30	DQ30	DQB14	DQB14		
DDR0_D43	B38	DQ31	DQ31	DQB15	DQB15		

Название вывода	Вывод на корпусе	DDR3	DDR4	LPDDR3	LPDDR4	Назначение	Домен питания
DDR0_D44	A40	DM3 x4: DQS_T12	DM3/ DBI3 x4: DQS_T12	DMB1	DMB1/ DBIB1	В режиме DDR3 вывод используется как маска записи данных (DM3), а при использовании микросхем с шириной шины данных x4 как строб записи данных (DQS_T12) В режиме DDR4 вывод используется как маска записи данных (DM3) или признак инверсии шины данных (DBI3), а при использовании микросхем с шириной шины данных x4 как строб записи данных (DQS_T12) В режиме LPDDR3 вывод используется как маска записи данных (DMB1) В режиме LPDDR4 вывод используется как маска записи данных (DMB1) или признак инверсии шины данных (DBIB1)	
DDR0_D45	A41	DQS_T3	DQS_T3	DQSB_T1	DQSB_T1	В режиме DDR3 и DDR4 вывод используется как строб записи данных DQS_*3 В режиме LPDDR3 и LPDDR4 вывод используется как строб записи данных DQSA_*3	
DDR0_D46	B41	DQS_C3	DQS_C3	DQSB_C1	DQSB_C1		
DDR0_D47	B40	NC x4: DQS_C12	NC x4: DQS_C12	NC	NC	Вывод используется как строб записи данных (DQS_C12) только в режимах DDR3 и DDR4 и только при использовании микросхем с шириной шины данных x4	
DDR0_A0	AA41	CKE0	CKE0	CKEA0	CKEA0	В режимах DDR3 и DDR4: • CKE* – разрешение подачи синхросигнала • CS_N* – выбор ранка • ACT_N – команда активации (только для DDR4) • BA* – выбор банка • BG* – выбор группы банков (только для DDR4) • A* – выбор адреса • CLK* – синхросигнал • PAR – контроль четности • CAS_N – выбор колонки • RAS_N – выбор строки • WE_N – маска записи	
DDR0_A1	AA42	CKE1	CKE1	CKEA1	CKEA1		
DDR0_A2	Y40	CS_N0	CS_N0	CSA0	CSA0		
DDR0_A3	Y41	NC	C0	CSA1	CSA1		
DDR0_A4	W41	BA2	BG0	CLKA_T	CLKA_T		
DDR0_A5	W42	A14	BG1	CLKA_C	CLKA_C		
DDR0_A6	V41	A15	ACT_N	NC	NC		
DDR0_A7	V42	A9	A9	NC	NC		
DDR0_A8	U41	A12	A12	CAA0	CAA0		
DDR0_A9	U42	A11	A11	CAA1	CAA1		
DDR0_A10	T40	A7	A7	CAA2	CAA2		
DDR0_A11	T41	A8	A8	CAA3	CAA3		
DDR0_A12	R41	A6	A6	CAA4	CAA4		
DDR0_A13	R42	A5	A5	CAA5	CAA5		

Название вывода	Вывод на корпусе	DDR3	DDR4	LPDDR3	LPDDR4	Назначение	Домен питания
DDR0_A14	P41	A4	A4	CAA6	NC	• C0, C1, C2 – идентификатор микросхемы (только для DDR4) • ODT* – выводы накристальной терминации В режимах LPDDR3 и LPDDR4: • CKEA*, CKEB* – разрешение подачи синхросигнала • CSA*, CSB* – выбор микросхемы • CLKA*, CLKB* – синхросигнал • CAA*, CAB* – шина команд и адреса • ODTA, ODTB – выводы накристальной терминации (только для LPDDR3)	
DDR0_A15	P42	A3	A3	CAA7	NC		
DDR0_A16	M41	CLK0_T	CLK0_T	CAA8	NC		
DDR0_A17	M42	CLK0_C	CLK0_C	CAA9	NC		
DDR0_A18	N41	NC	NC	ODTA	NC		
DDR0_A19	L42	Тестовый вывод					
DDR0_A20	K41	CLK1_T	CLK1_T	CKEB0	CKEB0		
DDR0_A21	K42	CLK1_C	CLK1_C	CKEB1	CKEB1		
DDR0_A22	L41	NC	NC	CSB1	CSB1		
DDR0_A23	J41	NC	NC	CSB0	CSB0		
DDR0_A24	H42	A2	A2	CLKB_T	CLKB_T		
DDR0_A25	H41	A1	A1	CLKB_C	CLKB_C		
DDR0_A26	J40	BA1	BA1	NC	NC		
DDR0_A27	H40	PAR	PAR	NC	NC		
DDR0_A28	G41	A13	A13	CAB0	CAB0		
DDR0_A29	G42	BA0	BA0	CAB1	CAB1		
DDR0_A30	F41	A10	A10	CAB2	CAB2		
DDR0_A31	E41	A0	A0	CAB3	CAB3		
DDR0_A32	E42	NC	C2	CAB4	CAB4		
DDR0_A33	D41	CAS_N	CAS_N	CAB5	CAB5		
DDR0_A34	C41	WE_N	WE_N	CAB6	NC		
DDR0_A35	C40	RAS_N	RAS_N	CAB7	NC		
DDR0_A36	D40	ODT0	ODT0	CAB8	NC		
DDR0_A37	D39	ODT1	ODT1	CAB9	NC		
DDR0_A38	D38	CS_N1	CS_N1	ODTB	NC		
DDR0_A39	C38	NC	C1	NC	NC		
DDR0_VREF	R40	VREF	VREF	VREF	VREF	Референсное напряжение	
DDR0_ZN	E38	ZN / ZN_SENSE	ZN / ZN_SENSE	ZN / ZN_SENSE	ZN / ZN_SENSE	Калибровочные вывод должен быть подключен к внешнему резистору номиналом 40, 120 или 240 Ом +/- 1%	

[а] Подтяжка выводов к земле или питанию выбирается на уровне контроллера DDRMC

[б] x4 – обозначат назначение вывода при работе с микросхемами с шириной данных 4 бита

51.1.2 Выводы DDRMC1

Таблица 51.3. Перечень сигнальных выводов. DDRMC1

Название вывода	Вывод на корпусе	DDR3	DDR4	LPDDR3	LPDDR4	Назначение	Домен питания
DDR1_MEM RESET_L	AY39	RESET_N	RESET_N	NC	RESET_N	Сигнал сброса	DDR1_VDDQ
DDR1_ALERT_N	AY40	NC	ALERT_N	NC	NC	Сигнализация ошибки от DRAM	
DDR1_D0	BB39	DQ0	DQ0	DQA0	DQA0	В режиме DDR3 и DDR4 вывод используется как шина данных DQ[7:0] В режиме LPDDR3 и LPDDR4 вывод используется как шина данных DQA[7:0]	
DDR1_D1	BD39	DQ1	DQ1	DQA1	DQA1		
DDR1_D2	BD38	DQ2	DQ2	DQA2	DQA2		
DDR1_D3	BC38	DQ3	DQ3	DQA3	DQA3		
DDR1_D4	BD44	DQ4	DQ4	DQA4	DQA4		
DDR1_D5	BD43	DQ5	DQ5	DQA5	DQA5		
DDR1_D6	BC42	DQ6	DQ6	DQA6	DQA6		
DDR1_D7	BD42	DQ7	DQ7	DQA7	DQA7		
DDR1_D8	BD41	DM0 x4: DQS_T9	DM0/ DBI0 x4: DQS_T9	DMA0	DMA0/ DBIA0	В режиме DDR3 вывод используется как маска записи данных (DM0), а при использовании микросхем с шириной шины данных x4 как строб записи данных (DQS_T9) В режиме DDR4 вывод используется как маска записи данных (DM0) или признак инверсии шины данных (DBI0), а при использовании микросхем с шириной шины данных x4 как строб записи данных (DQS_T9) В режиме LPDDR3 вывод используется как маска записи данных (DMA0) В режиме LPDDR4 вывод используется как маска записи данных (DMA0) или признак инверсии шины данных (DBIA0)	
DDR1_D9	BD40	DQS_T0	DQS_T0	DQSA_T0	DQSA_T0	В режиме DDR3 и DDR4 вывод используется как строб записи данных DQS_*0 В режиме LPDDR3 и LPDDR4 вывод используется как строб записи данных DQSA_*0	
DDR1_D10	BC40	DQS_C0	DQS_C0	DQSA_C0	DQSA_C0		
DDR1_D11	BB41	NC x4: DQS_C9	NC x4: DQS_C9	NC	NC	Вывод используется как строб записи данных (DQS_C9) только в режимах DDR3 и DDR4 и только	

Название вывода	Вывод на корпусе	DDR3	DDR4	LPDDR3	LPDDR4	Назначение	Домен питания
						при использовании микросхем с шириной шины данных x4	
DDR1_D12	BB44	DQ8	DQ8	DQA8	DQA8	В режиме DDR3 и DDR4 вывод используется как шина данных DQ[15:8] В режиме LPDDR3 и LPDDR4 вывод используется как шина данных DQA[15:8]	
DDR1_D13	BA42	DQ9	DQ9	DQA9	DQA9		
DDR1_D14	AV43	DQ10	DQ10	DQA10	DQA10		
DDR1_D15	AU42	DQ11	DQ11	DQA11	DQA11		
DDR1_D16	AU44	DQ12	DQ12	DQA12	DQA12		
DDR1_D17	AV44	DQ13	DQ13	DQA13	DQA13		
DDR1_D18	BB43	DQ14	DQ14	DQA14	DQA14		
DDR1_D19	BC44	DQ15	DQ15	DQA15	DQA15		
DDR1_D20	BA44	DM1 x4: DQS_T10	DM1/ DBI1 x4: DQS_T10	DMA1	DMA1/ DBIA1	В режиме DDR3 вывод используется как маска записи данных (DM1), а при использовании микросхем с шириной шины данных x4 как строб записи данных (DQS_T10) В режиме DDR4 вывод используется как маска записи данных (DM1) или признак инверсии шины данных (DBI1), а при использовании микросхем с шириной шины данных x4 как строб записи данных (DQS_T10) В режиме LPDDR3 вывод используется как маска записи данных (DMA1) В режиме LPDDR4 вывод используется как маска записи данных (DMA1) или признак инверсии шины данных (DBIA1)	
DDR1_D21	AY44	DQS_T1	DQS_T1	DQSA_T1	DQSA_T1	В режиме DDR3 и DDR4 вывод используется как строб записи данных DQS_*1 В режиме LPDDR3 и LPDDR4 вывод используется как строб записи данных DQSA_*1	
DDR1_D22	AW44	DQS_C1	DQS_C1	DQSA_C1	DQSA_C1		
DDR1_D23	AY43	NC x4: DQS_T10	NC x4: DQS_T10	NC	NC	Вывод используется как строб записи данных (DQS_C10) только в режимах DDR3 и DDR4 и только при использовании микросхем с шириной шины данных x4	
DDR1_D24	AL42	DQ16	DQ16	DQB0	DQB0	В режиме DDR3 и DDR4 вывод используется как шина данных DQ[23:16] В режиме LPDDR3 и LPDDR4 вывод используется как шина данных DQB[7:0]	
DDR1_D25	AR42	DQ17	DQ17	DQB1	DQB1		
DDR1_D26	AT44	DQ18	DQ18	DQB2	DQB2		
DDR1_D27	AT43	DQ19	DQ19	DQB3	DQB3		

Название вывода	Вывод на корпусе	DDR3	DDR4	LPDDR3	LPDDR4	Назначение	Домен питания
DDR1_D28	AR44	DQ20	DQ20	DQB4	DQB4		
DDR1_D29	AP43	DQ21	DQ21	DQB5	DQB5		
DDR1_D30	AL44	DQ22	DQ22	DQB6	DQB6		
DDR1_D31	AK43	DQ23	DQ23	DQB7	DQB7		
DDR1_D32	AM44	DM2 x4: DQS_T11	DM2/ DBI2 x4: DQS_T11	DMB0	DMB0/ DBIB0	В режиме DDR3 вывод используется как маска записи данных (DM2), а при использовании микросхем с шириной шины данных x4 как строб записи данных (DQS_T11) В режиме DDR4 вывод используется как маска записи данных (DM2) или признак инверсии шины данных (DBI2), а при использовании микросхем с шириной шины данных x4 как строб записи данных (DQS_T11) В режиме LPDDR3 вывод используется как маска записи данных (DMB0) В режиме LPDDR4 вывод используется как маска записи данных (DMB0) или признак инверсии шины данных (DBIB0)	
DDR1_D33	AN44	DQS_T2	DQS_T2	DQSB_T0	DQSB_T0	В режиме DDR3 и DDR4 вывод используется как строб записи данных DQS_*2 В режиме LPDDR3 и LPDDR4 вывод используется как строб записи данных DQSA_*2	
DDR1_D34	AP44	DQS_C2	DQS_C2	DQSB_C0	DQSB_C0		
DDR1_D35	AM43	NC x4: DQS_C11	NC x4: DQS_C11	NC	NC	Вывод используется как строб записи данных (DQS_C11) только в режимах DDR3 и DDR4 и только при использовании микросхем с шириной шины данных x4	
DDR1_D36	AH43	DQ24	DQ24	DQB8	DQB8	В режиме DDR3 и DDR4 вывод используется как шина данных DQ[31:24] В режиме LPDDR3 и LPDDR4 вывод используется как шина данных DQB[15:7]	
DDR1_D37	AJ44	DQ25	DQ25	DQB9	DQB9		
DDR1_D38	AJ42	DQ26	DQ26	DQB10	DQB10		
DDR1_D39	AK44	DQ27	DQ27	DQB11	DQB11		
DDR1_D40	AD44	DQ28	DQ28	DQB12	DQB12		
DDR1_D41	AD43	DQ29	DQ29	DQB13	DQB13		
DDR1_D42	AE42	DQ30	DQ30	DQB14	DQB14		
DDR1_D43	AE44	DQ31	DQ31	DQB15	DQB15		
DDR1_D44	AF44	DM3 x4: DQS_T12	DM3/ DBI3	DMB1	DMB1/ DBIB1	В режиме DDR3 вывод используется как маска записи данных (DM3), а при использовании микросхем с	

Название вывода	Вывод на корпусе	DDR3	DDR4	LPDDR3	LPDDR4	Назначение	Домен питания
			x4: DQS_T12			шириной шины данных x4 как строб записи данных (DQS_T12) В режиме DDR4 вывод используется как маска записи данных (DM3) или признак инверсии шины данных (DBI3), а при использовании микросхем с шириной шины данных x4 как строб записи данных (DQS_T12) В режиме LPDDR3 вывод используется как маска записи данных (DMB1) В режиме LPDDR4 вывод используется как маска записи данных (DMB1) или признак инверсии шины данных (DBIB1)	
DDR1_D45	AG44	DQS_T3	DQS_T3	DQSB_T1	DQSB_T1	В режиме DDR3 и DDR4 вывод используется как строб записи данных DQS_*3 В режиме LPDDR3 и LPDDR4 вывод используется как строб записи данных DQSA_*3	
DDR1_D46	AH44	DQS_C3	DQS_C3	DQSB_C1	DQSB_C1		
DDR1_D47	AF43	NC x4: DQS_C12	NC x4: DQS_C12	NC	NC	Вывод используется как строб записи данных (DQS_C12) только в режимах DDR3 и DDR4 и только при использовании микросхем с шириной шины данных x4	
DDR1_A0	BB40	CKE0	CKE0	CKEA0	CKEA0	В режимах DDR3 и DDR4: • CKE* – разрешение подачи синхросигнала • CS_N* – выбор ранка • ACT_N – команда активации (только для DDR4) • BA* – выбор банка • BG* – выбор группы банков (только для DDR4) • A* – выбор адреса • CLK* – синхросигнал • PAR – контроль четности • CAS_N – выбор колонки • RAS_N – выбор строки • WE_N – маска записи	
DDR1_A1	BA39	CKE1	CKE1	CKEA1	CKEA1		
DDR1_A2	BB38	CS_N0	CS_N0	CSA0	CSA0		
DDR1_A3	BA38	NC	C0	CSA1	CSA1		
DDR1_A4	BA41	BA2	BG0	CLKA_T	CLKA_T		
DDR1_A5	BA40	A14	BG1	CLKA_C	CLKA_C		
DDR1_A6	AY41	A15	ACT_N	NC	NC		
DDR1_A7	AY42	A9	A9	NC	NC		
DDR1_A8	AW42	A12	A12	CAA0	CAA0		
DDR1_A9	AW41	A11	A11	CAA1	CAA1		
DDR1_A10	AV41	A7	A7	CAA2	CAA2		
DDR1_A11	AV42	A8	A8	CAA3	CAA3		
DDR1_A12	AU41	A6	A6	CAA4	CAA4		
DDR1_A13	AU40	A5	A5	CAA5	CAA5		
DDR1_A14	AT41	A4	A4	CAA6	NC		
DDR1_A15	AT42	A3	A3	CAA7	NC		

Название вывода	Вывод на корпусе	DDR3	DDR4	LPDDR3	LPDDR4	Назначение	Домен питания
DDR1_A16	AR41	CLK0_T	CLK0_T	CAA8	NC	• C0, C1, C2 – идентификатор микросхемы (только для DDR4) • ODT* – выводы накристальной терминации В режимах LPDDR3 и LPDDR4: • CKEA*, CKEB* – разрешение подачи синхросигнала • CSA*, CSB* – выбор микросхемы • CLKA*, CLKB* – синхросигнал • CAA*, CAB* – шина команд и адреса • ODTA, ODTB – выводы накристальной терминации (только для LPDDR3)	
DDR1_A17	AR40	CLK0_C	CLK0_C	CAA9	NC		
DDR1_A18	AP41	NC	NC	ODTA	NC		
DDR1_A19	AP42	Тестовый вывод					
DDR1_A20	AN42	CLK1_T	CLK1_T	CKEB0	CKEB0		
DDR1_A21	AN41	CLK1_C	CLK1_C	CKEB1	CKEB1		
DDR1_A22	AM41	NC	NC	CSB1	CSB1		
DDR1_A23	AM42	NC	NC	CSB0	CSB0		
DDR1_A24	AL41	A2	A2	CLKB_T	CLKB_T		
DDR1_A25	AL40	A1	A1	CLKB_C	CLKB_C		
DDR1_A26	AK41	BA1	BA1	NC	NC		
DDR1_A27	AK42	PAR	PAR	NC	NC		
DDR1_A28	AJ40	A13	A13	CAB0	CAB0		
DDR1_A29	AJ41	BA0	BA0	CAB1	CAB1		
DDR1_A30	AG42	A10	A10	CAB2	CAB2		
DDR1_A31	AG41	A0	A0	CAB3	CAB3		
DDR1_A32	AH42	NC	C2	CAB4	CAB4		
DDR1_A33	AH41	CAS_N	CAS_N	CAB5	CAB5		
DDR1_A34	AF41	WE_N	WE_N	CAB6	NC		
DDR1_A35	AF42	RAS_N	RAS_N	CAB7	NC		
DDR1_A36	AE40	ODT0	ODT0	CAB8	NC		
DDR1_A37	AE41	ODT1	ODT1	CAB9	NC		
DDR1_A38	AD41	CS_N1	CS_N1	ODTB	NC		
DDR1_A39	AD42	NC	C1	NC	NC		
DDR1_VREF	AK40	VREF	VREF	VREF	VREF	Референсное напряжение	
DDR1_ZN	AY38	ZN / ZN_SENSE	ZN / ZN_SENSE	ZN / ZN_SENSE	ZN / ZN_SENSE	Калибровочные вывод должен быть подключен к внешнему резистору номиналом 40, 120 или 240 Ом +/- 1%	

[a)] Подтяжка выводов к земле или питанию выбирается на уровне контроллера DDRMC

51.1.3 Выводы NAND (GNSS_HV)

Таблица 51.4. Перечень сигнальных выводов. NAND (GNSS_HV)

Название вывода	Вывод на корпусе	Назначение NAND	Назначение GNSS	Тип вывода	Drive[б]	Pull up/down[б]	Состояние по сбросу	Домен питания
NFC_IO0	AP1	Шина данных NFC_IO[15:0] [а]	CMOS_HV_DIN[0]	I/O	4	Нет	Z	HSP_VDDO_NFC
NFC_IO1	AP5		CMOS_HV_DIN[1]				Z	
NFC_IO2	AP2		CMOS_HV_DIN[2]				Z	
NFC_IO3	AN4		CMOS_HV_DIN[3]				Z	
NFC_IO4	AN2		CMOS_HV_DIN[4]				Z	
NFC_IO5	AN6		CMOS_HV_DIN[5]				Z	
NFC_IO6	AN3		CMOS_HV_DIN[6]				Z	
NFC_IO7	AM3		CMOS_HV_DIN[7]				Z	
NFC_IO8	AN5		CMOS_HV_DIN[8]				Z	
NFC_IO9	AN1		CMOS_HV_DIN[9]				Z	
NFC_IO10	AM5		CMOS_HV_DIN[10]				Z	
NFC_IO11	AM2		CMOS_HV_DIN[11]				Z	
NFC_IO12	AM6		NC				Z	
NFC_IO13	AM4		NC				Z	
NFC_IO14	AL6		NC				Z	
NFC_IO15	AL5		NC				Z	
NFC_RBN0	AP6	Готовность/занятость памяти	NC	I	--	нет	-	
NFC_RBN1	AP4		NC				-	
NFC_ALE	AR1	Разрешение защелкивания адреса	NC	O	4	нет	Z	
NFC_CLE	AR4	Разрешение защелкивания команды	NC	O	4	нет	Z	
NFC_REN	AR3	Чтение	NC	O	4	нет	Z	
NFC_WEN	AP3	Запись	NC	O	4	нет	Z	
NFC_CEN0	AR5	Сигналы выбора блока памяти.	NC	O	4	нет	Z	
NFC_CEN1	AR2		NC				Z	
NFC_DQS	AM1	Сигнал строба данных	NC	I/O	4	нет	Z	

[а] КП используются совместно с блоком GNSS

[б] Указано значение по сбросу. Перенастраивается через HSPERIPH.URB.nand*_padcfg

51.1.4 Выводы QSPI0 и QSPI1

Таблица 51.5. Перечень сигнальных выводов. QSPI0

Название вывода	Вывод на корпусе	Назначение	Тип вывода	Drive	Pull up/down	Состояние по сбросу	Домен питания
QSPI0_SISO0	AW26	Линии данных	I/O	4	up	I	COREVDDPST
QSPI0_SISO1	AW24					I	
QSPI0_SISO2	AW25					I	
QSPI0_SISO3	AW23					I	
QSPI0_SS0	AV24	Выбор устройства	I/O	4	up	I	
QSPI0_SS1	AV23		O			O	
QSPI0_SS2	AV22		O			O	
QSPI0_SS3	AW22		O			O	
QSPI0_SCLK	AW27	Линия синхронизации	I/O	4	up	I	

Таблица 51.6. Перечень сигнальных выводов. QSPI1

Название вывода	Вывод на корпусе	Назначение	Тип вывода	Drive[a)]	Pull up/down [a)]	Состояние по сбросу	Домен питания
QSPI1_SISO0	AT2	Линии данных	I/O	4	нет	Z	HSP_VDDO_QSPI
QSPI1_SISO1	AT3					Z	
QSPI1_SISO2	AT4					Z	
QSPI1_SISO3	AT5					Z	
QSPI1_SS0	AR6	Выбор устройства	I/O	4	нет	Z	
QSPI1_SS1	AR7		O			Z	
QSPI1_SS2	AT6		O			Z	
QSPI1_SS3	AT7		O			Z	
QSPI1_SCLK	AT1	Линия синхронизации	I/O	4	нет	Z	

[a)] Указано значение по сбросу. Перенастраивается через HSPERIPH.URB.qspi*_padcfg

51.1.5 Выводы SDMMC0 и SDMMC1

Таблица 51.7. Перечень сигнальных выводов. SDMMC0 и SDMMC1

Название вывода	Вывод на корпусе	Назначение	Тип вывода	Drive [в]	Pull up/down	Состояние по сбросу	Домен питания
SDMMC0_CMD	BB3	Команда	I/O	4	нет [а]	Z	HSP_VDDO_SDMMC0
SDMMC0_DAT0	BC4	Шина данных	I/O	4	нет [а]	Z	HSP_VDDO_SDMMC0
SDMMC0_DAT1	BD2					Z	
SDMMC0_DAT2	BD3					Z	
SDMMC0_DAT3	BC2					Z	
SDMMC0_DAT4	BD4					Z	
SDMMC0_DAT5	BC3					Z	
SDMMC0_DAT6	BD1					Z	
SDMMC0_DAT7	BC1					Z	
SDMMC0_CLK	BB4	Тактовая частота	O	4	нет [а]	Z	HSP_VDDO_SDMMC0
SDMMC0_CDN	BB1	Определение наличия карты	I	--	up [в]	-	HSP_VDDO_MISC [б]
SDMMC0_WP	BB2	Защита от записи	I	--	up [в]	-	HSP_VDDO_MISC [б]
SDMMC0_18EN	BA2	Выбор рабочего напряжения	O	4	--	Z	HSP_VDDO_MISC [б]
SDMMC0_PWR	BA1	Включение питания карты памяти	O	4	--	Z	HSP_VDDO_MISC [б]
SDMMC1_CMD	AY1	Команда	I/O	4	нет [а]	Z	HSP_VDDO_SDMMC1
SDMMC1_DAT0	AW4	Шина данных	I/O	4	нет [а]	Z	HSP_VDDO_SDMMC1
SDMMC1_DAT1	AW3					Z	
SDMMC1_DAT2	AW1					Z	
SDMMC1_DAT3	AV2					Z	
SDMMC1_DAT4	AW2					Z	
SDMMC1_DAT5	AV3					Z	
SDMMC1_DAT6	AV1					Z	
SDMMC1_DAT7	AV4					Z	
SDMMC1_CLK	AY2	Тактовая частота	O	4	нет [а]	Z	HSP_VDDO_SDMMC1
SDMMC1_CDN	AU3	Определения наличия карты	I	--	up [в]	-	HSP_VDDO_MISC [б]
SDMMC1_WP	AU4	Защита от записи	I	--	up [в]	-	HSP_VDDO_MISC [б]
SDMMC1_18EN	AU2	Выбор рабочего напряжения	O	4	--	Z	HSP_VDDO_MISC [б]
SDMMC1_PWR	AU1	Включение питания карты памяти	O	4	--	Z	HSP_VDDO_MISC [б]

[а] Указано значение по сбросу. Перенастраивается через HSPERIPH.URB.sdmmc*_padcfg
[б] Используется также для питания некоторых КП USB
[в] Указано значение по сбросу. Перенастраивается через HSPERIPH.misc_padcfg

51.1.6 Выводы EMAC0 и EMAC1

Таблица 51.8. Перечень сигнальных выводов. EMAC0 и EMAC1

Название вывода	Вывод на корпусе	Назначение	Тип вывода	Drive [a)]	Pull up/down [a)]	Состояние по сбросу	Домен питания
CLK125	BD5	Тактовая частота 125 МГц	I	--	--	-	HSP_VDDO_EMAC
EMAC0_RGMII_MDIO	BC5	Данные по интерфейсу MD	I/O	4	нет	Z	
EMAC0_RGMII_MDC	BB5	Тактовая частота обмена данными по интерфейсу MD	O	4	нет	Z	
EMAC0_RGMII_TXC	AY9	Тактовая частота передачи данных	O	4	нет	Z	
EMAC0_RGMII_RXC	AY8	Тактовая частота приема данных	I	--	нет	-	
EMAC0_RGMII_TXD0	BD9	Шина передаваемых данных	O	4	нет	Z	
EMAC0_RGMII_TXD1	BC9					Z	
EMAC0_RGMII_TXD2	BB9					Z	
EMAC0_RGMII_TXD3	BA9					Z	
EMAC0_RGMII_RXD0	BD8	Шина принимаемых данных	I	--	нет	-	
EMAC0_RGMII_RXD1	BC8					-	
EMAC0_RGMII_RXD2	BB8					-	
EMAC0_RGMII_RXD3	BA8					-	
EMAC0_RGMII_TXCTL	AW9	Управляющий вывод передачи	O	4	нет	Z	
EMAC0_RGMII_RXCTL	AW8	Управляющий вывод приема	I	--	нет	-	
EMAC1_RGMII_MDIO	BA5	Данные по интерфейсу MD	I/O	4	нет	Z	
EMAC1_RGMII_MDC	AY5	Тактовая частота обмена данными по интерфейсу MD	O	4	нет	Z	
EMAC1_RGMII_TXC	AY7	Тактовая частота передачи данных	O	4	нет	Z	
EMAC1_RGMII_RXC	AY6	Тактовая частота приема данных	I	--	нет	-	
EMAC1_RGMII_TXD0	BD7	Шина передаваемых данных	O	4	нет	Z	
EMAC1_RGMII_TXD1	BC7					Z	
EMAC1_RGMII_TXD2	BB7					Z	
EMAC1_RGMII_TXD3	BA7					Z	
EMAC1_RGMII_RXD0	BD6	Шина принимаемых данных	I	--	нет	-	
EMAC1_RGMII_RXD1	BC6					-	
EMAC1_RGMII_RXD2	BB6					-	
EMAC1_RGMII_RXD3	BA6					-	
EMAC1_RGMII_TXCTL	AW7	Управляющий вывод передачи	O	4	нет	Z	

Название вывода	Вывод на корпусе	Назначение	Тип вывода	Drive [a)]	Pull up/down [a)]	Состояние по сбросу	Домен питания
EMAC1_RGMII_RXCTL	AW6	Управляющий вывод приема	I	--	нет	-	

[a)] Указано значение по сбросу. Перенастраивается через HSPERIPH.URB.emac*_padcfg

51.1.7 Выводы USB0 и USB1

Таблица 51.9. Перечень сигнальных выводов. USB0 и USB1

Название вывода	Вывод на корпусе	Назначение	Тип вывода	Drive	Pull up/down	Состояние по сбросу	Домен питания
USB0_ID0	AY13	Не используется. Оставить не подключенным.	I	--	--	-	0 – 1.8 В
USB0_VBUS0	AY15	Подключается к источнику VBUS USB (во всех режимах).	I/O	--	--	-	0 – 5.25 В
USB0_DP0	BD12	Данные (прямой)	I/O diff	--	--	-	
USB0_DM0	BC12	Данные (инверсный)	I/O diff	--	--	-	
USB0_RESREF	AW15	Подключение калибровочного резистора (200 Ом±1%)	--	--	--	-	
USB0_REF_CLK_P	BA12	Опорная частота (прямая), 100 МГц	I/O diff			-	
USB0_REF_CLK_M	AY12	Опорная частота (инверсная), 100 МГц	I/O diff			-	
USB0_RX0_P	BD13	Данные приема в режиме superspeed (прямой)	I/O diff			-	
USB0_RX0_M	BC13	Данные приема в режиме superspeed (инверсный)	I/O diff			-	
USB0_TX0_P	BB13	Данные передачи в режиме superspeed (прямой)	I/O diff			-	
USB0_TX0_M	BA13	Данные передачи в режиме superspeed (инверсный)	I/O diff			-	
USB0_EN_OCN	AW12	Управление питанием VBUS/Превышение тока	I/O [a)]	4 [a)]	--	0	HSP_VDDO_MISC [б)]
USB1_ID0	AY11	Не используется. Оставить не подключенным.	I	--	--	-	0 – 1.8 В
USB1_VBUS0	AY14	Подключается к источнику VBUS USB (во всех режимах).	I/O	--	--	-	0 – 5.25 В
USB1_DP0	BD10	Данные (прямой)	I/O diff	--	--	-	
USB1_DM0	BC10	Данные (инверсный)	I/O diff	--	--	-	
USB1_RESREF	AW14	Подключение калибровочного резистора (200 Ом±1%)	--	--	--	-	
USB1_REF_CLK_P	BA10	Опорная частота (прямая), 100 МГц	I/O diff			-	
USB1_REF_CLK_M	AY10	Опорная частота (инверсная), 100 МГц	I/O diff			-	
USB1_RX0_P	BD11	Данные приема в режиме superspeed (прямой)	I/O diff			-	
USB1_RX0_M	BC11	Данные приема в режиме superspeed (инверсный)	I/O diff			-	

USB1_TX0_P	BB11	Данные передачи в режиме superspeed (прямой)	I/O diff			-	
USB1_TX0_M	BA11	Данные передачи в режиме superspeed (инверсный)	I/O diff			-	
USB1_EN_OCN	AW11	Управление питанием VBUS/Превышение тока	I/O [а]	4 [а]	--	0	HSP_VDDO_MISC [б]

[а] Работает в режиме Open-Drain. Ток КП перенастраивается через HSPERIPH.URB.misc_padcfg

[б] Используется также для питания некоторых КП SDMMC

51.1.8 Выводы для видео ввода/вывода

Таблица 51.10. Перечень сигнальных выводов. Видео ввод/вывод

Выводы параллельной шины видео ввода/вывода мультиплексируются между портами ввода и вывода. Направление выводов управляется битом DISPLAY_PARALLEL_PORT_EN регистра SUBSYSTEM_CFG мультимедийной подсистемы (адрес 0x12_2000). Если DISPLAY_PARALLEL_PORT_EN=0, выводы PIXCLK, HSYNC, VSYNC и VDATA0-11 являются входами (дефолтное значение после сброса), если DISPLAY_PARALLEL_PORT_EN=1 – перечисленные выводы являются выходами, на выводы VDATA выводятся сигналы данных интерфейса с дисплеем процессора дисплея.

Название вывода	Вывод на корпусе	Функция 0	Функция 1	Назначение	Тип вывода	Drive	Pull up/down	Состояние по сбросу	Домен питания
PIXCLK	D25			Синхросигнал параллельных портов ввода/вывода видео	I/O	8	Down	I	MEDIA_VDDPST
HSYNC	A25			Сигнал горизонтальной синхронизации	I/O	8	down	I	
VSYNC	C25			Сигнал вертикальной синхронизации	I/O	8	down	I	
DE	B25			Сигнал разрешения передачи видеоданных	I/O	8	down	I	
VDATA0	C24	Вход ISP	PXLDATAB0[4]		I/O	8	down	I	
VDATA1	A21		PXLDATAB0[5]		I/O	8	down	I	
VDATA2	D21		PXLDATAB0[6]		I/O	8	down	I	
VDATA3	A22		PXLDATAB0[7]		I/O	8	down	I	
VDATA4	F22		PXLDATAB0[8]		I/O	8	down	I	
VDATA5	B21		PXLDATAB0[9]		I/O	8	down	I	
VDATA6	D22		PXLDATAB0[10]		I/O	8	down	I	
VDATA7	A23		PXLDATAB0[11]		I/O	8	down	I	
VDATA8	D23		PXLDATAG0[4]		I/O	8	down	I	
VDATA9	B22		PXLDATAG0[5]		I/O	8	down	I	
VDATA10	D24		PXLDATAG0[6]		I/O	8	down	I	
VDATA11	C21		PXLDATAG0[7]		I/O	8	down	I	
VDATA12	E22		PXLDATAG0[8]		I/O	8	down	I	

Название вывода	Вывод на корпусе	Функция 0	Функция 1	Назначение	Тип вывода	Drive	Pull up/down	Состояние по сбросу	Домен питания
VDATA13	B23	-	PXLDATAG0[9]	Шина видеоданных	I/O	8	down	I	
VDATA14	F23		PXLDATAG0[10]		I/O	8	down	I	
VDATA15	A24		PXLDATAG0[11]		I/O	8	down	I	
VDATA16	E23		PXLDATAR0[4]		I/O	8	down	I	
VDATA17	C23		PXLDATAR0[5]		I/O	8	down	I	
VDATA18	E24		PXLDATAR0[6]		I/O	8	down	I	
VDATA19	C22		PXLDATAR0[7]		I/O	8	down	I	
VDATA20	F24		PXLDATAR0[8]		I/O	8	down	I	
VDATA21	E25		PXLDATAR0[9]		I/O	8	down	I	
VDATA22	F25		PXLDATAR0[10]		I/O	8	down	I	
VDATA23	B24		PXLDATAR0[11]		I/O	8	down	I	
CMOS0_CLK	A19			Синхросигнал CMOS сенсора 0	O	8	down	O	
CMOS0_FSYNC	B19			Сигнал кадровой синхронизации CMOS сенсора 0	O	8	down	O	
CMOS1_CLK	C19			Синхросигнал CMOS сенсора 1	O	8	down	O	
CMOS1_FSYNC	D19			Сигнал кадровой синхронизации CMOS сенсора 1	O	8	down	O	
VTEST0	E20			Шина тестирования ISP (для внутреннего применения)	O	8	down	O	
VTEST1	B20				O	8	down	O	
VTEST2	F20				O	8	down	O	
VTEST3	A20				O	8	down	O	
VTEST4	F21				O	8	down	O	
VTEST5	D20				O	8	down	O	
VTEST6	E21				O	8	down	O	
VTEST7	C20				O	8	down	O	

Таблица 51.11 Перечень сигнальных выводов. Видео MIPI DSI, MIPI CSI

Название вывода	Вывод на корпусе	Назначение	Тип вывода
MIPI_RX0_CLKP	F29	Сигнал синхронизации (прямой) порта MIPI RX 0	IO diff
MIPI_RX0_CLKN	E29	Сигнал синхронизации (инверсный) порта MIPI RX 0	IO diff
MIPI_RX0_DATAP0	C29	Сигнал data lane 0 (прямой) порта MIPI RX 0	IO diff
MIPI_RX0_DATAN0	B29	Сигнал data lane 0 (инверсный) порта MIPI RX 0	IO diff
MIPI_RX0_DATAP1	B30	Сигнал data lane 1 (прямой) порта MIPI RX 0	IO diff

Название вывода	Вывод на корпусе	Назначение	Тип вывода
MIPI_RX0_DATAN1	A30	Сигнал data lane 1 (инверсный) порта MIPI RX 0	IO diff
MIPI_RX0_DATAP2	C31	Сигнал data lane 2 (прямой) порта MIPI RX 0	IO diff
MIPI_RX0_DATAN2	B31	Сигнал data lane 2 (инверсный) порта MIPI RX 0	IO diff
MIPI_RX0_DATAP3	F31	Сигнал data lane 3 (прямой) порта MIPI RX 0	IO diff
MIPI_RX0_DATAN3	E31	Сигнал data lane 3 (инверсный) порта MIPI RX 0	IO diff
MIPI_RX1_CLKP	E32	Сигнал синхронизации (прямой) порта MIPI RX 1	IO diff
MIPI_RX1_CLKN	D32	Сигнал синхронизации (инверсный) порта MIPI RX 1	IO diff
MIPI_RX1_DATAP0	B32	Сигнал data lane 0 (прямой) порта MIPI RX 1	IO diff
MIPI_RX1_DATAN0	A32	Сигнал data lane 0 (инверсный) порта MIPI RX 1	IO diff
MIPI_RX1_DATAP1	C33	Сигнал data lane 1 (прямой) порта MIPI RX 1	IO diff
MIPI_RX1_DATAN1	B33	Сигнал data lane 1 (инверсный) порта MIPI RX 1	IO diff
MIPI_RX1_DATAP2	B34	Сигнал data lane 2 (прямой) порта MIPI RX 1	IO diff
MIPI_RX1_DATAN2	A34	Сигнал data lane 2 (инверсный) порта MIPI RX 1	IO diff
MIPI_RX1_DATAP3	E34	Сигнал data lane 3 (прямой) порта MIPI RX 1	IO diff
MIPI_RX1_DATAN3	D34	Сигнал data lane 3 (инверсный) порта MIPI RX 1	IO diff
MIPI_TX_CLKP	D28	Сигнал синхронизации (прямой) порта MIPI TX	IO diff
MIPI_TX_CLKN	E28	Сигнал синхронизации (инверсный) порта MIPI TX	IO diff
MIPI_TX_DATAP0	B28	Сигнал data lane 0 (прямой) порта MIPI TX	IO diff
MIPI_TX_DATAN0	A28	Сигнал data lane 0 (инверсный) порта MIPI TX	IO diff
MIPI_TX_DATAP1	C27	Сигнал data lane 1 (прямой) порта MIPI TX	IO diff
MIPI_TX_DATAN1	B27	Сигнал data lane 1 (инверсный) порта MIPI TX	IO diff
MIPI_TX_DATAP2	B26	Сигнал data lane 2 (прямой) порта MIPI TX	IO diff
MIPI_TX_DATAN2	A26	Сигнал data lane 2 (инверсный) порта MIPI TX	IO diff
MIPI_TX_DATAP3	D26	Сигнал data lane 3 (прямой) порта MIPI TX	IO diff
MIPI_TX_DATAN3	E26	Сигнал data lane 3 (инверсный) порта MIPI TX	IO diff
MIPI_RX0_REXT	D30	Внешний референсный резистор MIPI RX0	A [a)]
MIPI_RX1_REXT	E33	Внешний референсный резистор MIPI RX1	A [a)]
MIPI_TX_REXT	E27	Внешний референсный резистор MIPI TX	A [a)]
MIPI_RX0_ATB	E30	Аналоговый тестовый вывод MIPI RX0	A
MIPI_RX1_ATB	F33	Аналоговый тестовый вывод MIPI RX1	A
MIPI_TX_ATB	F27	Аналоговый тестовый вывод MIPI TX	A

[a] - к выводам должен подключаться резистор номиналом 200 Ом

51.1.9 Выводы MFBSP0 и MFBSP1

Таблица 51.12. Перечень сигнальных выводов. MFBSP0 и MFBSP1

Название вывода	Вывод на корпусе	Функция 0 [a]	Функция 1 [a]	Назначение	Тип вывода	Drive	Pull up/down	Состояние по сбросу	Домен питания
MFBSP0_LDAT0	BB24			Шина данных	I/O	8	up	I	COREVDDPST
MFBSP0_LDAT1	BA24				I/O	8	up	I	
MFBSP0_LDAT2	BC24				I/O	8	up	I	
MFBSP0_LDAT3	AY24				I/O	8	up	I	
MFBSP0_LDAT4	BD24				I/O	8	up	I	
MFBSP0_LDAT5	BA23				I/O	8	up	I	
MFBSP0_LDAT6	BB23	MFBSP0_LDAT6	RX_CAN0 [b]		I/O	8	up	I	
MFBSP0_LDAT7	AY23	MFBSP0_LDAT7	TX_CAN0 [b]		I/O	8	up	I	
MFBSP0_LCLK	BC23			Синхронизация данных	I/O	8	down	I	COREVDDPST
MFBSP0_LACK	BB22			Подтверждение приема данных	I/O	8	down	I	COREVDDPST
MFBSP1_LDAT0	BD23			Шина данных	I/O	8	up	I	COREVDDPST
MFBSP1_LDAT1	BA22				I/O	8	up	I	
MFBSP1_LDAT2	BC22				I/O	8	up	I	
MFBSP1_LDAT3	BB21				I/O	8	up	I	
MFBSP1_LDAT4	BD22				I/O	8	up	I	
MFBSP1_LDAT5	BA21				I/O	8	up	I	
MFBSP1_LDAT6	BC21	MFBSP1_LDAT6	RX_CAN1 [b]		I/O	8	up	I	
MFBSP1_LDAT7	AY22	MFBSP1_LDAT7	TX_CAN1 [b]		I/O	8	up	I	
MFBSP1_LCLK	BD21			Синхронизация данных	I/O	8	down	I	COREVDDPST
MFBSP1_LACK	AY21			Подтверждение приема данных	I/O	8	down	I	COREVDDPST

a) Все выводы MFBSP могут выполнять различные функции в зависимости от настроек портов MFBSP0 и MFBSP1. Подробнее см. Главу 37.

b) Дополнительно выводы LDAT6 (как вход) и LDAT7 (как выход) могут использоваться как RX и TX выводы CAN. Подробнее см. Главу 37.

51.1.10 Служебные выводы

Таблица 51.13. Перечень сигнальных выводов. Служебные выводы

Название вывода	Вывод на корпусе	Назначение	Тип вывода	Drive	Pull up/down	Состояние по сбросу	Домен питания
I2C4_SCL	BD25	Линия синхронизации	I/O	8	up	I	COREVDDPST
I2C4_SDA	BC25	Линия данных	I/O	8	up	I	COREVDDPST
I2C4_CUR_PU_EN	BB25	Сигнал включения токовой подтяжки	O	8	up	O	COREVDDPST
XTI274_XIN	BD20	Вход осциллятора 27.456 МГц – опорная частота микросхемы	I OSC	*	*	I	COREVDDPST
XTI274_XOUT	BC20	Выход осциллятора 27.456 МГц – опорная частота микросхемы	O OSC	*	*	O	COREVDDPST
TDO	AY27	Выход данных теста (JTAG)	O	*	*	O	COREVDDPST
TCK	BD27	Тестовый тактовый сигнал (JTAG)	I	4	up	I	COREVDDPST
TRSTN	BC27	Установка исходного состояния (JTAG)	I	4	up	I	COREVDDPST
TDI	BA27	Вход данных теста (JTAG)	I	4	up	I	COREVDDPST
TMS	BB27	Выбор режима теста (JTAG)	I	4	up	I	COREVDDPST
TESTMODE	AY28	Режим тестирования и отладки микросхемы	I	4	down	I	COREVDDPST
JMODE0	BB28	При TESTMODE = 0; 0x0 – RISC0 JTAG; 0x1 USOC USB0; 0x2 USOC JTAG; 0x3 Зарезервировано При TESTMODE =1; 0x0 BoundaryScan; 0x1 Scan; 0x2 MBIST; 0x3 Зарезервировано	I	4	down	I	COREVDDPST
JMODE1	BA28		I	4	down	I	COREVDDPST
SERV_SPARE1	BB30	Переводит микросхему в режим сканирования: 0: Функциональный режим 1: Режим сканирования	I	4	up	I	COREVDDPST
BOOT0	BD29	Источник данных при начальной загрузке микросхемы и ядра RISC0, после снятия сигнала сброса	I	4	down	I	COREVDDPST
BOOT1	BD28		I	4	down	I	COREVDDPST
BOOT2	BC28		I	4	down	I	COREVDDPST
VS_EN	AY29	Отключает разграничения по уровням доступа secure, trusted, sdr	I	4	down	I	COREVDDPST
BS_EN	AW29	Задание необходимости авторизации образа загрузчика при начальной загрузке	I	4	down	I	COREVDDPST
CPU_OFFREQN	BA25	Запрос на выключение домена CPU (AVDD)	O	4	up	O	COREVDDPST
MEDIA_OFFREQN	AY26	Запрос на выключение домена MEDIA (MVDD)	O	4	up	O	COREVDDPST

Название вывода	Вывод на корпусе	Назначение	Тип вывода	Drive	Pull up/down	Состояние по сбросу	Домен питания
SDR_OFFREQN	BD26	Запрос на выключение домена SDR (SVDD)	O	4	up	O	COREVDDPST
CPU_OFFACKN	AY25	Подтверждение выключения домена CPU (AVDD)	I	4	down	I	COREVDDPST
MEDIA_OFFACKN	BA26	Подтверждение выключения домена MEDIA (MVDD)	I	4	down	I	COREVDDPST
SDR_OFFACKN	BC26	Подтверждение выключения домена SDR (SVDD)	I	4	down	I	COREVDDPST
EXTINT	BB29	Линия внешних прерываний	I/O	4	down	I	COREVDDPST
			I/O	-	-	-	VPP
			I/O	-	-	-	VQQ
CLKOUT	BB26	Универсальный вывод синхросигнала	O	4	up	O	COREVDDPST
NRST_WRM (SPARE0)	BB20	Сигнал сброса микросхемы, кроме RTC таймера	I	4	up	I	COREVDDPST
CORE_OFFREQN	BC31	Запрос на выключение домена CORE (CVDD)	O	8	up	O	BAT
XTI32K_XIN	BD30	Вход/Выход осциллятора 32.768 КГц – опорная частота RTC и домена BAT	I OSC	*	*	I	BAT
XTI32K_XOUT	BC30	Вход/Выход осциллятора 32.768 КГц – опорная частота RTC и домена BAT	O OSC	*	*	O	BAT
NRST_PON	BD31	Сигнал сброса микросхемы; также сбрасывает RTC	I	4	up	I	BAT
TS_VCAL1[1)]	C37	Выводы для калибровки накристальных датчиков температуры	A	*	*	*	MEDIA_TAVDD
TS_VSENSE1[1)]	D37		A	*	*	*	MEDIA_TAVDD

[1)] Входной аналоговый пин TS_VCAL и выходной аналоговый пин TS_VSENSE подключены к датчику температуры. В нормальном режиме работы эти выводы не используются и нужны только для режима калибровки. Если калибровка не нужна, выводы можно оставить неподключенными.

51.1.11 Выводы GPIO и периферийных устройств подсистем LSPERIPH0 и LSPERIPH1

Функциональное назначения каждого вывода подсистем lsperiph0_subs и lsperiph1_subs задается соответствующим битом соответствующего регистра gpio_swport*_ctl блока GPIO подсистемы. Подробное описание в Главе 36 «Контроллер универсальных выводов (GPIO)».

Таблица 51.14. Сигнальные выводы. Периферийные устройства LSPERIPH0

Название вывода	Вывод на корпусе	Функция 0	Назначение 0	Функция 1	Назначение 1	Тип вывода	Drive	Pull up/down	Состояние по сбросу	Домен питания
GPIO0_PORTA_0	BB36	GPIO0_PORTA_0 [a)]	Порт A GPIO0	UART3_OUT1_N	Модемные программируемые сигналы	I/O	4	up	I	COREVDDPST
GPIO0_PORTA_1	BB37	GPIO0_PORTA_1 [a)]		UART3_OUT2_N	Модемные программируемые сигналы	I/O	4	up	I	
GPIO0_PORTA_2	AW37	GPIO0_PORTA_2 [a)]		UART3_CTS_N	Модемные сигналы статуса	I/O	4	up	I	
GPIO0_PORTA_3	AY37	GPIO0_PORTA_3 [a)]		UART3_DSR_N		I/O	4	up	I	
GPIO0_PORTA_4	BD36	GPIO0_PORTA_4 [a)]		UART3_DCD_N		I/O	4	up	I	
GPIO0_PORTA_5	BC36	GPIO0_PORTA_5 [a)]		UART3_RI_N		I/O	4	up	I	
GPIO0_PORTA_6	BA37	GPIO0_PORTA_6 [a)]		UART3_DTR_N		I/O	4	up	I	
GPIO0_PORTA_7	AW36	GPIO0_PORTA_7 [a)]		UART3_RTS_N		I/O	4	up	I	
GPIO0_PORTB_0	BC35	GPIO0_PORTB_0	Порт B GPIO0	UART3_SIN	Вход последовательных данных	I/O	4	up	I	
GPIO0_PORTB_1	BB35	GPIO0_PORTB_1		UART3_SOUT	Выход последовательных данных	I/O	4	up	I	
GPIO0_PORTB_2	AY36	GPIO0_PORTB_2		UART3_DE	Сигнал состояния передатчика RS485	I/O	4	up	I	
GPIO0_PORTB_3	BA36	GPIO0_PORTB_3		UART3_RE	Сигнал состояния приемника RS485	I/O	4	up	I	

Название вывода	Вывод на корпусе	Функция 0	Назначение 0	Функция 1	Назначение 1	Тип вывода	Drive	Pull up/down	Состояние по сбросу	Домен питания
GPIO0_PORTB_4	BB34	GPIO0_PORTB_4		UART3_RS485_EN	Сигнал включения RS485	I/O	4	up	I	
GPIO0_PORTB_5	BD35	GPIO0_PORTB_5		UART1_SIN	Вход последовательных данных	I/O	4	up	I	
GPIO0_PORTB_6	AW35	GPIO0_PORTB_6		UART1_SOUT	Выход последовательных данных	I/O	4	up	I	
GPIO0_PORTB_7	AY35	GPIO0_PORTB_7		UART2_SIN	Вход последовательных данных	I/O	4	up	I	
GPIO0_PORTC_0	BD34	GPIO0_PORTC_0	Порт C GPIO0	SSI0_SCLK_OUT	Выходной тактовый синхросигнал	I/O	4	up	I	
GPIO0_PORTC_1	BC34	GPIO0_PORTC_1		SSI0_TXD	Выход последовательных данных	I/O	4	up	I	
GPIO0_PORTC_2	BA35	GPIO0_PORTC_2		SSI0_RXD	Вход последовательных данных	I/O	4	up	I	
GPIO0_PORTC_3	AY34	GPIO0_PORTC_3		SSI0_SS_IN	Сигнал арбитража в режиме multi-mater	I/O	4	up	I	
GPIO0_PORTC_4	BC33	GPIO0_PORTC_4		SSI0_SS_0	Выходной сигнал выбора slave-устройства	I/O	4	up	I	
GPIO0_PORTC_5	BB33	GPIO0_PORTC_5		SSI0_SS_1		I/O	4	up	I	
GPIO0_PORTC_6	BA34	GPIO0_PORTC_6		SSI0_SS_2		I/O	4	up	I	
GPIO0_PORTC_7	AW34	GPIO0_PORTC_7		SSI0_SS_3		I/O	4	up	I	

Название вывода	Вывод на корпусе	Функция 0	Назначение 0	Функция 1	Назначение 1	Тип вывода	Drive	Pull up/down	Состояние по сбросу	Домен питания
GPIO0_PORTD_0	BB32	GPIO0_PORTD_0	Порт D GPIO0	UART2_SOUT	Выход последовательных данных	I/O	4	up	I	
GPIO0_PORTD_1	BD33	GPIO0_PORTD_1		UART2_SIR_IN	Вход последовательных данных IrDA	I/O	4	up	I	
GPIO0_PORTD_2	AY33	GPIO0_PORTD_2		UART2_SIR_OUT_N	Выход последовательных данных IrDA	I/O	4	up	I	
GPIO0_PORTD_3	BA33	GPIO0_PORTD_3		I2C0_SCL	Последовательная линия тактирования	I/O	4	up	I	
GPIO0_PORTD_4	BD32	GPIO0_PORTD_4		I2C0_SDA	Последовательная линия данных	I/O	4	up	I	
GPIO0_PORTD_5	BC32	GPIO0_PORTD_5		I2C0_SMBALERT	Вход прерывания SMBALERT интерфейса SMBUS.	I/O	4	up	I	
GPIO0_PORTD_6	AW33	GPIO0_PORTD_6		I2C0_SMBSUS_IN_N	Входной сигнал управления low power suspend mode.	I/O	4	up	I	
GPIO0_PORTD_7	BA32	GPIO0_PORTD_7		I2C0_SMBSUS_OUT_N	Выходной сигнал управления low power suspend mode.	I/O	4	up	I	

a) Вывод можно использовать в качестве входа прерывания. Подробнее см. Главу 10.

Каждая КП подсистемы lsperiph1_subs имеет свой конфигурационный регистр управления GPIO1_PORTx_n_PAD_CTR. Для более подробной информации см. раздел 11.2.

Параметры каждой КП могут независимо настраиваться в следующих диапазонах:

1. Тип вывода: может быть I/O. После сброса КП работает как Output.
2. Pullup/down: направление подтяжки может быть Pull Up, Pull Down, Week Pull. После сброса все подтяжки выключены.
3. Drive: настраивается в диапазоне от 2 до 12 mA с шагом в 2 mA. По сбросу: 6 mA.

Таблица 51.15. Сигнальные выводы. Периферийные устройства LSPERIPH1

Название вывода	Вывод на корпусе	Функция 0		Функция 1	Назначение	Тип вывода	Drive	Pull up/down	Состояние по сбросу	Домен питания
GPIO1_PORTA_0	BB16	GPIO1_PORTA_0 [a]	Порт A GPIO1	I2C1_SCL	Последовательная линия тактирования	I/O	6	--	z	LSP1_VDDO
GPIO1_PORTA_1	BB19	GPIO1_PORTA_1 [a)]		I2C1_SDA	Последовательная линия данных	I/O	6	--	z	
GPIO1_PORTA_2	BB17	GPIO1_PORTA_2 [a)]		I2C2_SCL	Последовательная линия тактирования	I/O	6	--	z	
GPIO1_PORTA_3	BC19	GPIO1_PORTA_3 [a)]		I2C2_SDA	Последовательная линия данных	I/O	6	--	z	
GPIO1_PORTA_4	BB14	GPIO1_PORTA_4 [a)]		I2C3_SCL	Последовательная линия тактирования	I/O	6	--	z	
GPIO1_PORTA_5	BD19	GPIO1_PORTA_5 [a)]		I2C3_SDA	Последовательная линия данных	I/O	6	--	z	
GPIO1_PORTA_6	BB15	GPIO1_PORTA_6 [a)]		-		I/O	6	---	z	
GPIO1_PORTA_7	BA17	GPIO1_PORTA_7 [a)]		-		I/O	6	--	z	
GPIO1_PORTB_0	BB18	GPIO1_PORTB_0	Порт B GPIO1	I2S0_SCLK_OUT	Тактовый синхросигнал	I/O	6	--	z	
GPIO1_PORTB_1	BA18	GPIO1_PORTB_1		I2S0_WS_OUT	Сигнал кадровой синхронизации	I/O	6	--	z	
GPIO1_PORTB_2	BD18	GPIO1_PORTB_2		I2S0_SDO0	Последовательная линия 0 выдачи аудиоданных	I/O	6	--	z	

Название вывода	Вывод на корпусе	Функция 0		Функция 1	Назначение	Тип вывода	Drive	Pull up/down	Состояние по сбросу	Домен питания
GPIO1_PORTB_3	BA15	GPIO1_PORTB_3		I2S0_SDO1	Последовательная линия 1 выдачи аудиоданных	I/O	6	--	z	
GPIO1_PORTB_4	BC18	GPIO1_PORTB_4		I2S0_SDI0	Последовательная линия 0 приема аудиоданных	I/O	6	--	z	
GPIO1_PORTB_5	BA16	GPIO1_PORTB_5		I2S0_SCLK_IN	Дополнительный опорный источник синхросигнала	I/O	6	--	z	
GPIO1_PORTB_6	AY18	GPIO1_PORTB_6		UART0_SOUT	Выход последовательных данных	I/O	6	--	z	
GPIO1_PORTB_7	BD17	GPIO1_PORTB_7		UART0_SIN	Вход последовательных данных	I/O	6	--	z	
GPIO1_PORTC_0	BA14	GPIO1_PORTC_0	Порт C GPIO1	SSI1_SCLK_OUT	Тактовый синхросигнал	I/O	6	--	z	
GPIO1_PORTC_1	BC17	GPIO1_PORTC_1		SSI1_TXD	Выход последовательных данных	I/O	6	--	z	
GPIO1_PORTC_2	AY16	GPIO1_PORTC_2		SSI1_RXD	Вход последовательных данных	I/O	6	--	z	
GPIO1_PORTC_3	BD16	GPIO1_PORTC_3		SSI1_SS_0_N	Выбор slave-устройства	I/O	6	--	z	
GPIO1_PORTC_4	AY17	GPIO1_PORTC_4		SSI1_SS_1_N		I/O	6	--	z	
GPIO1_PORTC_5	AW18	GPIO1_PORTC_5		SSI1_SS_2_N		I/O	6	--	z	
GPIO1_PORTC_6	BD15	GPIO1_PORTC_6		SSI1_SS_3_N		I/O	6	--	z	
GPIO1_PORTC_7	AW19	GPIO1_PORTC_7		SSI1_SS_IN_N	Входной сигнал slave_select.	I/O	6	--	z	
GPIO1_PORTD_0	BC16	GPIO1_PORTD_0	Порт D GPIO1	TIMERS_TOGGLE_0	ШИМ выход 0 таймера	I/O	6	--	z	

Название вывода	Вывод на корпусе	Функция 0		Функция 1	Назначение	Тип вывода	Drive	Pull up/down	Состояние по сбросу	Домен питания
GPIO1_PORTD_1	AW16	GPIO1_PORTD_1		TIMERS_TOGGLE_1	ШИМ выход 1 таймера	I/O	6	--	z	
GPIO1_PORTD_2	BC15	GPIO1_PORTD_2		PWM_OENA0	ШИМ каналы	I/O	6	--	z	
GPIO1_PORTD_3	AW17	GPIO1_PORTD_3		PWM_OENB0		I/O	6	--	z	
GPIO1_PORTD_4	AV19	GPIO1_PORTD_4		PWM_OENA1		I/O	6	--	z	
GPIO1_PORTD_5	BC14	GPIO1_PORTD_5		PWM_OENB1		I/O	6	--	z	
GPIO1_PORTD_6	AV18	GPIO1_PORTD_6		PWM_TU[0]	Размерность рабочего хода (Trip Unit) для ШИМ	I/O	6	--	z	
GPIO1_PORTD_7	BD14	GPIO1_PORTD_7		PWM_TU[1]		I/O	6	--	z	

51.1.12 Выводы SDR подсистемы для подключения GPIO/ЦАП/АЦП

Таблица 51.16. Сигнальные выводы. Универсальные CMOS выводы SDR для подключения GPIO/SPI/ADC

Название	Вывод на корпусе	Функция 0	Функция 1	Функция 2	Функция 3	Тип вывода	Drive	Pull up/down	Состояние по сбросу	Домен питания
DFE_GPIO0	D9	Dfe_gpio0 (IO)	Dfe_spi1_clk (O)	Adc_cmos_data0 (I)	Adc_cmos_data0 (I)	I/O	8	down	I	SDR_VDDPST
DFE_GPIO1	C9	Dfe_gpio1 (IO)	Dfe_spi1_mosi (O)	Adc_cmos_data1 (I)	Adc_cmos_data1 (I)	I/O	8	down	I	
DFE_GPIO2	F9	Dfe_gpio2 (IO)	Dfe_spi1_miso (I)	Adc_cmos_data2 (I)	Adc_cmos_data2 (I)	I/O	8	down	I	
DFE_GPIO3	F8	Dfe_gpio3 (IO)	Dfe_spi1_cs1 (O)	Adc_cmos_data3 (I)	Adc_cmos_data3 (I)	I/O	8	down	I	
DFE_GPIO4	C8	Dfe_gpio4 (IO)	Dfe_spi1_cs2 (O)	Adc_cmos_data4 (I)	Adc_cmos_data4 (I)	I/O	8	down	I	
DFE_GPIO5	C7	Dfe_gpio5 (IO)	Dfe_spi1_cs4 (O)	Adc_cmos_data5 (I)	Adc_cmos_data5 (I)	I/O	8	down	I	
DFE_GPIO6	F7	Dfe_gpio6 (IO)	Dfe_spi1_cs8 (O)	Adc_cmos_data6 (I)	Adc_cmos_data6 (I)	I/O	8	down	I	
DFE_GPIO7	E9	Dfe_gpio7 (IO)	Dfe_spi2_clk (O)	Adc_cmos_data7 (I)	Adc_cmos_data7 (I)	I/O	8	down	I	
DFE_GPIO8	B8	Dfe_gpio8 (IO)	Dfe_spi2_mosi (O)	Adc_cmos_data8 (I)	Adc_cmos_data8 (I)	I/O	8	down	I	
DFE_GPIO9	B7	Dfe_gpio9 (IO)	Dfe_spi2_miso (I)	Adc_cmos_data9 (I)	Adc_cmos_data9 (I)	I/O	8	down	I	
DFE_GPIO10	E8	Dfe_gpio10 (IO)	Dfe_spi2_cs1 (O)	Adc_cmos_data10 (I)	Adc_cmos_data10 (I)	I/O	8	down	I	
DFE_GPIO11	E7	Dfe_gpio11 (IO)	Dfe_spi2_cs2 (O)	Adc_cmos_data11 (I)	Adc_cmos_data11 (I)	I/O	8	down	I	
DFE_GPIO12	A7	Dfe_gpio12 (IO)	Dfe_spi2_cs4 (O)	Adc_cmos_data12 (I)	Adc_cmos_data12 (I)	I/O	8	down	I	

Название	Вывод на корпусе	Функция 0	Функция 1	Функция 2	Функция 3	Тип вывода	Drive	Pull up/down	Состояние по сбросу	Домен питания
DFE_GPIO13	A8	Dfe_gpio13 (IO)	Dfe_spi2_cs8 (O)	Adc_cmos_data13 (I)	Adc_cmos_data13 (I)	I/O	8	down	I	
DFE_GPIO14	D8	Dfe_gpio14 (IO)	Dfe_gpio14 (IO)	Adc_cmos_data14 (I)	Adc_cmos_data14 (I)	I/O	8	down	I	
DFE_GPIO15	D7	Dfe_gpio15 (IO)	Dfe_gpio15 (IO)	Adc_cmos_data15 (I)	Adc_cmos_data15 (I)	I/O	8	down	I	
DFE_GPIO16	U5	Dfe_gpio16 (IO)	Dfe_spi3_clk (O)	Dfe_spi3_clk (O)	Gnss_gpio0 (IO)	I/O	8	down	I	
DFE_GPIO17	U6	Dfe_gpio17 (IO)	Dfe_spi3_mosi (O)	Dfe_spi3_mosi (O)	Gnss_gpio1 (IO)	I/O	8	down	I	
DFE_GPIO18	U1	Dfe_gpio18 (IO)	Dfe_spi3_miso (I)	Dfe_spi3_miso (I)	Gnss_gpio2 (IO)	I/O	8	down	I	
DFE_GPIO19	U2	Dfe_gpio19 (IO)	Dfe_spi3_cs1 (O)	Dfe_spi3_cs1 (O)	Gnss_gpio3 (IO)	I/O	8	down	I	
DFE_GPIO20	U3	Dfe_gpio20 (IO)	Dfe_spi3_cs2 (O)	Dfe_spi3_cs2 (O)	Gnss_gpio4 (IO)	I/O	8	down	I	
DFE_GPIO21	U4	Dfe_gpio21 (IO)	Dfe_spi3_cs4 (O)	Dfe_spi3_cs4 (O)	Gnss_gpio5 (IO)	I/O	8	down	I	
DFE_GPIO22	T2	Dfe_gpio22 (IO)	Dfe_spi3_cs8 (O)	Dfe_spi3_cs8 (O)	Gnss_gpio6 (IO)	I/O	8	down	I	
DFE_GPIO23	T3	Dfe_gpio23 (IO)	Dfe_spi4_clk (O)	Dfe_spi4_clk (O)	Gnss_gpio7 (IO)	I/O	8	down	I	
DFE_GPIO24	T5	Dfe_gpio24 (IO)	Dfe_spi4_mosi (O)	Dfe_spi4_mosi (O)	Dfe_alt_tst[0] (O)	I/O	8	down	I	
DFE_GPIO25	T6	Dfe_gpio25 (IO)	Dfe_spi4_miso (I)	Dfe_spi4_miso (I)	Dfe_alt_tst[1] (O)	I/O	8	down	I	
DFE_GPIO26	R3	Dfe_gpio26 (IO)	Dfe_spi4_cs1 (O)	Dfe_spi4_cs1 (O)	Dfe_alt_tst[2] (O)	I/O	8	down	I	
DFE_GPIO27	T1	Dfe_gpio27 (IO)	Dfe_spi4_cs2 (O)	Dfe_spi4_cs2 (O)	Dfe_alt_tst[3] (O)	I/O	8	down	I	
DFE_GPIO28	R4	Dfe_gpio28 (IO)	Dfe_spi4_cs4 (O)	Dfe_spi4_cs4 (O)	Dfe_alt_tst[4] (O)	I/O	8	down	I	
DFE_GPIO29	T4	Dfe_gpio29 (IO)	Dfe_spi4_cs8 (O)	Dfe_spi4_cs8 (O)	Dfe_alt_tst[5] (O)	I/O	8	down	I	
DFE_GPIO30	R1	Dfe_gpio30 (IO)	Dfe_gpio30 (IO)	Adc_cmos_frame (I)	Adc_cmos _frame (I)	I/O	8	down	I	
DFE_GPIO31	R2	Dfe_gpio31 (IO)	Dfe_gpio31 (IO)	Dfe_gpio31 (IO)	Dfe_alt_tx_mark (O)	I/O	8	down	I	

Контактные площадки DFE_GPIO* можно использовать для следующих целей:

- В режиме «Функция 0» все 32 КП работают в режиме GPIO под управлением блока DFE_NELS
- В режиме «Функция 1» также все КП работают под управлением блока DFE_NELS, но большинство из них используются для подключения внешних устройств по SPI интерфейсу. Реализовано 4 независимых SPI с возможность подключения до 4-х подканалов (до 15, если к портам *cs* подключить внешний дешифратор 1:2:4:8 в 16):
    - DFE_GPIO[6:0] – SPI1
    - DFE_GPIO[13:7] – SPI2

- DFE_GPIO[15:14] – в режиме GPIO
- DFE_GPIO[22:16] – SPI3
- DFE_GPIO[29:23] – SPI4
- DFE_GPIO[31:30] – в режиме GPIO

- В режиме «Функция 2» часть КП отдается для подключения 16-разрядного АЦП (с управляющим сигналом “frame”), при этом сохраняются два интерфейса SPI:
  - DFE_GPIO[15:0] – АЦП data (заводится на порты {dfe_nels_adc5_d*,dfe_nels_adc4_d*} блока DFE_NELS и порт dfe_alt_adc_d* блока DFE_ALT)
  - DFE_GPIO[30] – АЦП frame (заводится на порт dfe_nels_adc4_frame блока DFE_NELS)
  - DFE_GPIO[22:16] – SPI3
  - DFE_GPIO[29:23] – SPI4
  - DFE_GPIO[31] – в режиме GPIO

- В режиме «Функция 3» вместо интерфейсов SPI на КП выводятся отладочные сигналы от блоков DFE_ALT и GNSS:
  - DFE_GPIO[15:0] – АЦП data (заводится на порты {dfe_nels_adc5_d*,dfe_nels_adc4_d*} блока DFE_NELS и порт dfe_alt_adc_d* блока DFE_ALT)
  - DFE_GPIO[30] – АЦП frame (заводится на порт dfe_nels_adc4_frame блока DFE_NELS)
  - DFE_GPIO[23:16] – в режиме GNSS_GPIO (под управлением блока GNSS)
  - DFE_GPIO[29:24] – DFE_ALT_TST
  - DFE_GPIO[31] – DFE_ALT_TX_MARK

Таблица 51.17. Сигнальные выводы. LVDS выводы для подключения к АЦП/ЦАП

Название	Вывод на корпусе	Назначение	Тип вывода	Состояние по сбросу	Домен питания
DFE_ADC1_D0_PAD	N4	Шина данных Dfe_adc1_d[7:0]	I	I	SDR_VDDO_R_RX
DFE_ADC1_D0_PADX	N3				

Название	Вывод на корпусе	Назначение	Тип вывода	Состояние по сбросу	Домен питания
DFE_ADC1_D1_PAD	N2		I	I	
DFE_ADC1_D1_PADX	N1				
DFE_ADC1_D2_PAD	M4		I	I	
DFE_ADC1_D2_PADX	M3				
DFE_ADC1_D3_PAD	M2		I	I	
DFE_ADC1_D3_PADX	M1				
DFE_ADC1_D4_PAD	L4		I	I	
DFE_ADC1_D4_PADX	L3				
DFE_ADC1_D5_PAD	L2		I	I	
DFE_ADC1_D5_PADX	L1				
DFE_ADC1_D6_PAD	K4		I	I	
DFE_ADC1_D6_PADX	K3				
DFE_ADC1_D7_PAD	K2		I	I	
DFE_ADC1_D7_PADX	K1				
DFE_ADC1_CLK_PAD	J4	Синхронизация данных dfe_adc1_clk	I	I	
DFE_ADC1_CLK_PADX	J3				
DFE_ADC1_FRAME_PAD	J2	Синхронизация фреймов dfe_adc1_frame	I	I	
DFE_ADC1_FRAME_PADX	J1				
DFE_ADC2_D0_PAD	H2	Шина данных Dfe_adc2_d[7:0]	I	I	
DFE_ADC2_D0_PADX	H1				
DFE_ADC2_D1_PAD	G4		I	I	
DFE_ADC2_D1_PADX	G3				
DFE_ADC2_D2_PAD	G2		I	I	
DFE_ADC2_D2_PADX	G1				
DFE_ADC2_D3_PAD	F2		I	I	
DFE_ADC2_D3_PADX	F1				
DFE_ADC2_D4_PAD	E2		I	I	
DFE_ADC2_D4_PADX	E1				
DFE_ADC2_D5_PAD	D2		I	I	
DFE_ADC2_D5_PADX	D1				
DFE_ADC2_D6_PAD	C2		I	I	
DFE_ADC2_D6_PADX	C1				
DFE_ADC2_D7_PAD	B2		I	I	
DFE_ADC2_D7_PADX	B1				
DFE_ADC2_CLK_PAD	A2	Синхронизация данных dfe_adc2_clk	I	I	

Название	Вывод на корпусе	Назначение	Тип вывода	Состояние по сбросу	Домен питания
DFE_ADC2_CLK_PADX	A1				
DFE_ADC2_FRAME_PAD	E3	Синхронизация фреймов dfe_adc2_frame (при работе с LVDS ADC) или SYNC_B синхронизация от DAC к Jesd1 (при работе с JESD204B DAC)	I	I	
DFE_ADC2_FRAME_PADX	F3				
DFE_DAC1_D0_PAD	AL2	Шина данных Dfe_dac1_d[15:0]	O	O	SDR_VDDO_TX
DFE_DAC1_D0_PADX	AL1				
DFE_DAC1_D1_PAD	AL4		O	O	
DFE_DAC1_D1_PADX	AL3				
DFE_DAC1_D2_PAD	AK2		O	O	
DFE_DAC1_D2_PADX	AK1				
DFE_DAC1_D3_PAD	AK4		O	O	
DFE_DAC1_D3_PADX	AK3				
DFE_DAC1_D4_PAD	AJ2		O	O	
DFE_DAC1_D4_PADX	AJ1				
DFE_DAC1_D5_PAD	AJ4		O	O	
DFE_DAC1_D5_PADX	AJ3				
DFE_DAC1_D6_PAD	AH2		O	O	
DFE_DAC1_D6_PADX	AH1				
DFE_DAC1_D7_PAD	AH4		O	O	
DFE_DAC1_D7_PADX	AH3				
DFE_DAC1_D8_PAD	AG2		O	O	
DFE_DAC1_D8_PADX	AG1				
DFE_DAC1_D9_PAD	AG4		O	O	
DFE_DAC1_D9_PADX	AG3				
DFE_DAC1_D10_PAD	AF2		O	O	
DFE_DAC1_D10_PADX	AF1				
DFE_DAC1_D11_PAD	AF4		O	O	
DFE_DAC1_D11_PADX	AF3				
DFE_DAC1_D12_PAD	AE2		O	O	
DFE_DAC1_D12_PADX	AE1				
DFE_DAC1_D13_PAD	AE4		O	O	
DFE_DAC1_D13_PADX	AE3				
DFE_DAC1_D14_PAD	AD2		O	O	

Название	Вывод на корпусе	Назначение	Тип вывода	Состояние по сбросу	Домен питания
DFE_DAC1_D14_PADX	AD1				
DFE_DAC1_D15_PAD	AD4		O	O	
DFE_DAC1_D15_PADX	AD3				
DFE_DAC1_CLK_PAD	AC1	Синхронизация данных	I	I	SDR_VDDO_R_RX
DFE_DAC1_CLK_PADX	AC2				

Таблица 51.18. Сигнальные выводы. CMOS выводы для подключения к ЦАП

Название	Вывод на корпусе	Назначение	Тип вывода	Состояние по сбросу	Drive	Pull Up/ Down	Домен питания
DFE_DAC3_D0	Y3	Шина данных Dfe_dac3_d[15:0]	O	O	8	down	SDR_VDDPST
DFE_DAC3_D1	Y4		O	O	8	down	
DFE_DAC3_D2	Y1		O	O	8	down	
DFE_DAC3_D3	Y2		O	O	8	down	
DFE_DAC3_D4	W5		O	O	8	down	
DFE_DAC3_D5	W6		O	O	8	down	
DFE_DAC3_D6	W2		O	O	8	down	
DFE_DAC3_D7	W3		O	O	8	down	
DFE_DAC3_D8	V6		O	O	8	down	
DFE_DAC3_D9	W4		O	O	8	down	
DFE_DAC3_D10	V3		O	O	8	down	
DFE_DAC3_D11	W1		O	O	8	down	
DFE_DAC3_D12	V4		O	O	8	down	
DFE_DAC3_D13	V5		O	O	8	down	
DFE_DAC3_D14	V1		O	O	8	down	
DFE_DAC3_D15	V2		O	O	8	down	
DFE_DAC3_CLK	Y5	Синхронизация данных Dfe_dac3_d	I	I	8	down	

Таблица 51.19. Сигнальные выводы. Дополнительные выводы DFE

Название	Вывод на корпусе	Назначение	Тип	Drive	Pull Up/Down	Состояние по сбросу	Домен питания
DFE_CLK_PAD	A5	Синхронизация DFE_NELS	I			I	SDR_VDDO_T_RX
DFE_CLK_PADX	B5						
DFE_TIMER	E6	Синхронизация таймера	I	8	down	I	SDR_VDDPST
DFE_PPS	Y6	Секундная метка DFE	I	8	down	I	

DFE_ADC4_CLK	F6	Синхронизация данных adc_cmos_data	I	8	down	I	

Таблица 51.20. Служебные LVDS IO*

Название	Вывод на корпусе	Назначение	Соответствие функциональным КП
SEG1_REF0_PAD	N5	Референсное напряжение (1.2 В)	DFE_ADC1*_PAD* DFE_ADC2*_PAD* JESD0_CLK_PAD* JESD1_CLK_PAD* SPARE_RX_PAD* DFE_DAC1_CLK_PAD*
SEG1_REF1_PAD	M5		
SEG1_REF2_PAD	L5		
SEG1_REF3_PAD	K5		
SEG1_REF4_PAD	J5		
SEG1_REF5_PAD	H5		
SEG1_RTTPAD_PAD	H4	Вывод для подключения к питанию SDR_VDDO_R_RX через внешнее сопротивление номиналом 100 Ом (точность 1%)	
SEG1_RTTPAD_PADX	H3	Вывод для подключения к земле через внешнее сопротивление номиналом 100 Ом	
SEG2_REF0_PAD	AK5	Референсное напряжение (1.2 В)	DFE_DAC1_D*_PAD* JESD_SYNCB_O_PAD* SPARE_TX_PAD*
SEG2_REF1_PAD	AJ5		
SEG2_REF2_PAD	AH5		
SEG2_REF3_PAD	AG5		
SEG2_REF4_PAD	AF5		
SEG3_REF0_PAD	E5	Референсное напряжение (1.2 В)	DFE_CLK_PAD* JESD0_SYSREF_PAD* JESD1_SYSREF_PAD*
SEG3_RTTPAD_PAD	C5	Вывод для подключения к питанию SDR_VDDO_T_RX через внешнее сопротивление номиналом 100 Ом (точность 1%)	
SEG3_RTTPAD_PADX	D5	Вывод для подключения к земле через внешнее сопротивление номиналом 100 Ом	

*Все функциональные LVDS IO разделены на три сегмента (SEG1, SEG2, SEG3), служебные LVDS IO нужны для корректной работы функциональных LVDS IO находящихся в соответствующем сегменте. В столбце таблицы «Соответствие функциональным КП» представлен состав функциональных IO каждого сегмента.

51.1.13 Выводы GNSS

Таблица 51.21. Сигнальные выводы. CMOS выводы GNSS

Название	Вывод на корпусе	Назначение	Тип	Drive	Pull Up/Down	Состояние по сбросу	Домен питания
GNSS_CLK	AC3	Синхронизация данных	I	8	Down	I	SDR_VDDPST
GNSS_D0	AB5	Шина данных gnss_d[11:0]	I	8	Down	I	
GNSS_D1	AB6		I	8	Down	I	

Название	Вывод на корпусе	Назначение	Тип	Drive	Pull Up/Down	Состояние по сбросу	Домен питания
GNSS_D2	AB2		I	8	Down	I	
GNSS_D3	AB3		I	8	Down	I	
GNSS_D4	AA6		I	8	Down	I	
GNSS_D5	AB4		I	8	Down	I	
GNSS_D6	AA3		I	8	Down	I	
GNSS_D7	AB1		I	8	Down	I	
GNSS_D8	AA4		I	8	Down	I	
GNSS_D9	AA5		I	8	Down	I	
GNSS_D10	AA1		I	8	Down	I	
GNSS_D11	AA2		I	8	Down	I	
GNSS_PPS	AC4	Секундная метка	O	8	-	O	

51.1.14 Выводы PCIe0, PCIe1 и JESD204B

Таблица 51.22. Сигнальные выводы. Выводы PCIe0 и PCIe1

Название вывода	Вывод на корпусе	Назначение	Тип вывода	Drive	Pull up / down	Состояние по сбросу	Домен питания
PCI0_PERSTN	H12	Фундаментальный сброс контроллера со слота PCI0	I/O	8	up	I	SDR_VDDPST
PCI0_WAKE	G12	Запрос на выход из режима пониженного энергопотребления в слот PCI0	I/O	8		O	
PCI0_APP_LTSSM_EN	G14	Разрешение начала процедуры инициализации контроллеров PCI0	I	8	down	I	
PCI0_CLKREQ	G13	Отключение внешнего генератора формирующего опорную частоту PCI0	I/O	8		O	
PCI0_PHY_RESREF	A9	Вывод для подключения калибровочного резистора номиналом 200 Ом ± 1% для PCI0	I/O			I	
PCI0_REF_CLK_M	A14	Опорная частота физического уровня PCI0, 100 МГц	IO diff			I	
PCI0_REF_CLK_P	B14		IO diff			I	
PCI0_RXN0	B13	Линии приема данных PCI0	IO diff			I	
PCI0_RXP0	C13		IO diff			I	
PCI0_RXN1	A12		IO diff			I	
PCI0_RXP1	B12		IO diff			I	

Название вывода	Вывод на корпусе	Назначение	Тип вывода	Drive	Pull up / down	Состояние по сбросу	Домен питания
PCI0_RXN2	E13		IO diff			I	
PCI0_RXP2	F13		IO diff			I	
PCI0_RXN3	D12		IO diff			I	
PCI0_RXP3	E12		IO diff			I	
PCI0_TXN0	B11	Линии передачи данных PCI0	IO diff			O	
PCI0_TXP0	C11		IO diff			O	
PCI0_TXN1	A10		IO diff			O	
PCI0_TXP1	B10		IO diff			O	
PCI0_TXN2	E11		IO diff			O	
PCI0_TXP2	F11		IO diff			O	
PCI0_TXN3	D10		IO diff			O	
PCI0_TXP3	E10		IO diff			O	
PCI1_PERSTN	H16	Фундаментальный сброс контроллера со слота PCI1	I/O	8	up	I	SDR_VDDPST
PCI1_WAKE	G16	Запрос на выход из режима пониженного энергопотребления в слот PCI1	I/O	8		O	
PCI1_APP_LTSSM_EN	H14	Разрешение начала процедуры инициализации контроллеров PCI1	I	8	down	I	
PCI1_CLKREQ	G15	Отключение внешнего генератора формирующего опорную частоту PCI1	I/O	8		O	
PCI1_PHY_RESREF	B9	Вывод для подключения калибровочного резистора номиналом 200 Ом ± 1% для PCI1	I/O			I	
PCI1_REF_CLK_M	E14	Опорная частота физического уровня PCI1, 100 МГц	IO diff			I	
PCI1_REF_CLK_P	D14		IO diff			I	
PCI1_RXN0	A18	Линии передачи данных PCI1	IO diff			I	
PCI1_RXP0	B18		IO diff			I	
PCI1_RXN1	B17		IO diff			I	
PCI1_RXP1	C17		IO diff			I	
PCI1_RXN2	D18		IO diff			I	

Название вывода	Вывод на корпусе	Назначение	Тип вывода	Drive	Pull up / down	Состояние по сбросу	Домен питания
PCI1_RXP2	E18		IO diff			I	
PCI1_RXN3	E17		IO diff			I	
PCI1_RXP3	F17		IO diff			I	
PCI1_TXN0	A16	Линии приема данных PCI1	IO diff			O	
PCI1_TXP0	B16		IO diff			O	
PCI1_TXN1	B15		IO diff			O	
PCI1_TXP1	C15		IO diff			O	
PCI1_TXN2	D16		IO diff			O	
PCI1_TXP2	E16		IO diff			O	
PCI1_TXN3	E15		IO diff			O	
PCI1_TXP3	F15		IO diff			O	

Таблица 51.23 Сигнальные выводы. Выводы JESD204B

Название вывода	Вывод на корпусе	Назначение	Тип вывода	Drive	Pull up / down	Состояние по сбросу	Домен питания
JESD0_CLK_PAD	B4	Синхронизация JESD204b (DEV_CLK)	I			I	SDR_VDDO_R_RX
JESD0_CLK_PADX	A4						
JESD0_SYSREF_PAD	A6	Синхронизация от генератора частоты dev_clk	I			I	SDR_VDDO_T_RX
JESD0_SYSREF_PADX	B6						
SPARE_RX_PAD	F4	Синхронизация от DAC к Jesd0 (SYNC_B_I)	I			I	SDR_VDDO_R_RX
SPARE_RX_PADX	E4						
SPARE_TX_PAD	C4	Синхронизация от Jesd0 к ADC (SYNC_B_O)	O			O	SDR_VDDO_TX
SPARE_TX_PADX	D4						
JESD1_CLK_PAD	B3	Синхронизация JESD204b (DEV_CLK)	I			I	SDR_VDDO_R_RX
JESD1_CLK_PADX	A3						
JESD1_SYSREF_PAD	C6	Синхронизация от генератора частоты dev_clk	I			I	SDR_VDDO_T_RX

Название вывода	Вывод на корпусе	Назначение	Тип вывода	Drive	Pull up / down	Состояние по сбросу	Домен питания
JESD1_SYSREF_PADX	D6						
JESD_SYNCB_O_PAD	C3	Синхронизация от Jesd1 к ADC (SYNC_B_O)	O			O	SDR_VDDO_TX
JESD_SYNCB_O_PADX	D3						

В микросхеме реализована возможность работы с различными устройствами через одни и те же КП. В данном случае речь идет об устройствах с интерфейсами PCIe и Jesd204b. Возможны следующие варианты подключения устройств:

- два с интерфейсом PCIe
- одно с интерфейсом PCIe, другое с Jesd204b
- два с интерфейсом Jesd204b

Более подробная схема подключения к внешним устройствам представлена в Главе 52 «Справочная информация».

51.2 Выводы питания

Таблица 51.24. Общее питание и питание домена CORE

Название выводов	Вывод на корпусе	Назначение	Объединяемые в корпусе линии питания	Номиналы напряжений, В	Число выводов
VSS	A11, A13, A15, A17, A27, A29, A31, A33, A37, AA11, AA12, AA13, AA17, AA18, AA19, AA23, AA24, AA25, AA29, AA30, AA31, AA32, AA33, AA36, AA37, AB10, AB14, AB15, AB16, AB20, AB21, AB22, AB26, AB27, AB30,	Глобальная цифровая земля		0	

Название выводов	Вывод на корпусе	Назначение	Объединяемые в корпусе линии питания	Номиналы напряжений, В	Число выводов
	AB31, AB34, AB35, AB38, AB39, AB40, AB41, AB7, AB8, AB9, AC10, AC14, AC15, AC16, AC20, AC21, AC22, AC26, AC27, AC30, AC31, AC34, AC35, AC38, AC39, AC40, AC41, AC5, AC6, AC7, AC9, AD11, AD12, AD13, AD17, AD18, AD19, AD23, AD24, AD28, AD29, AD32, AD33, AD36, AD37, AD5, AD6, AD7, AE11, AE12, AE13, AE17, AE18, AE19, AE23, AE24, AE25, AE28, AE29, AE32, AE33, AE36, AE37, AE43, AE5, AF10, AF14, AF15, AF16, AF20, AF21, AF22, AF23, AF24, AF25, AF28, AF29, AF32, AF33, AF36, AF37, AF8, AG10, AG14, AG15, AG16, AG20, AG21, AG22, AG25, AG26, AG27, AG30, AG31, AG34, AG35, AG38, AG39, AG40, AG43, AG8, AG9, AH10, AH11, AH12, AH13, AH17, AH18, AH19,				

Название выводов	Вывод на корпусе	Назначение	Объединяемые в корпусе линии питания	Номиналы напряжений, В	Число выводов
	AH25, AH26, AH27, AH30, AH31, AH34, AH35, AH38, AH39, AH40, AH6, AH7, AH8, AH9, AJ11, AJ12, AJ13, AJ17, AJ18, AJ19, AJ30, AJ31, AJ34, AJ35, AJ43, AJ6, AJ7, AK14, AK15, AK16, AK20, AK21, AK22, AK23, AK24, AK28, AK29, AK32, AK33, AK36, AK37, AK6, AK7, AL10, AL14, AL15, AL16, AL20, AL21, AL22, AL23, AL24, AL28, AL29, AL32, AL33, AL36, AL37, AL43, AL9, AM10,AM11, AM12, AM13, AM17, AM18, AM21, AM22, AM25, AM26, AM27, AM30, AM31, AM34, AM35, AM38, AM39, AM40, AM9, AN12, AN13, AN17, AN18, AN21, AN22, AN25, AN26, AN27, AN30, AN31, AN34, AN35, AN38, AN39, AN40, AN43, AN7, AN8, AP13, AP14, AP18, AP19, AP20, AP23, AP24, AP30, AP31, AP34,				

Название выводов	Вывод на корпусе	Назначение	Объединяемые в корпусе линии питания	Номиналы напряжений, В	Число выводов
	AP35, AP7, AP8, AP9, AR10, AR13, AR14, AR18, AR19, AR20, AR23, AR24, AR28, AR29, AR32, AR33, AR36, AR37, AR43, AR9, AT10, AT11, AT12, AT13, AT14, AT15, AT16, AT17, AT21, AT22, AT28, AT29, AT32, AT33, AT36, AT37, AT38, AT39, AT8, AT9, AU14, AU15, AU16, AU17, AU21, AU22, AU25, AU26, AU29, AU30, AU33, AU34, AU37, AU38, AU39, AU43, AU9, AV14, AV15, AV25, AV26, AV29, AV30, AV33, AV34, AV37, AV9, AW10, AW13, AW43, AW5, AY30, B37, B39, B42, BA43, BB10, BB12, BB42, BC37, BC39, BC41, BC43, BD37, C10, C12, C14, C16, C18, C26, C28, C30, C32, C34, D11, D13, D15, D17, D27, D29, D31, D33, D43, F10, F12, F14, F16, F18, F26, F28, F30, F32, F34, F43, G19, G20, G23,				

Название выводов	Вывод на корпусе	Назначение	Объединяемые в корпусе линии питания	Номиналы напряжений, В	Число выводов
	G24, G27, G29, G33, G37, G5, G6, G8, G9, H19, H20, H23, H24, H28, H29, H30, H34, H35, H38, H39, H43, H6, H8, H9, J10, J13, J14, J17, J18, J21, J22, J23, J24, J28, J29, J30, J34, J35, J38, J39, J8, J9, K10, K13, K14, K17, K18, K21, K22, K26, K27, K28, K32, K33, K36, K37, K43, K8, K9, L10, L13, L14, L17, L18, L21, L22, L26, L27, L28, L32, L33, L36, L37, L43, L8, L9, M11, M12, M15, M16, M19, M20, M23, M24, M25, M29, M30, M31, M34, M35, M38, M39, M40, M6, M7, N11, N12, N15, N16, N19, N20, N23, N24, N25, N29, N30, N31, N34, N35, N38, N39, N40, N43, N6, N7, P10, P13, P14, P17, P18, P21, P22, P26, P27, P28, P32, P33, P36, P37, P7, P8, P9, R10, R13, R14, R17, R18, R21, R22, R26, R27, R28, R32, R33, R36, R37, R43, R7, R8, R9, T11, T12,				

Название выводов	Вывод на корпусе	Назначение	Объединяемые в корпусе линии питания	Номиналы напряжений, В	Число выводов
	T13, T17, T18, T19, T23, T24, T25, T29, T30, T31, T34, T35, U11, U12, U13, U17, U18, U19, U23, U24, U25, U29, U30, U31, U34, U35, U38, U39, U40, U43, V10, V14, V15, V16, V20, V21, V22, V26, V27, V28, V34, V35, V38, V39, V40, V43, V7, V8, V9, W10, W14, W15, W16, W20, W21, W22, W26, W27, W28, W32, W33, W36, W37, W7, W8, W9, Y11, Y12, Y13, Y17, Y18, Y19, Y23, Y24, Y25, Y29, Y30, Y32, Y33, Y36, Y37, Y43				
CVDD	M32, M33, M36, M37, N32, N33, N36, N37, P34, P35, R34, R35, T32, T33, T36, T37, U32, U33, U36, U37, V32, V33, V36,V37, W34, W35, Y34, Y35, AA34, AA35, AB32, AB33, AB36, AB37, AC32, AC33, AC36, AC37, AD34, AD35, AE34, AE35, AF34, AF35, AG23, AG24, AG28, AG29, AG32,	Напряжение питания домена питания PD_CORE		0.9	

Название выводов	Вывод на корпусе	Назначение	Объединяемые в корпусе линии питания	Номиналы напряжений, В	Число выводов
	AG33, AG36, AG37, AH23, AH24, AH28, AH29, AH32, AH33, AH36, AH37, AJ8, AJ9, AJ10, AJ23, AJ24, AJ32, AJ33, AJ36, AJ37, AK8, AK9, AK10, AK11, AK12, AK13, AL11, AL12, AL13, AM23, AM24, AM36, AM37, AN9, AN10, AN11, AN23, AN24, AN36, AN37, AP10, AP11, AP12, AP15, AP16, AP17, AP21, AP22, AP36, AP37, AR11, AR12, AR15, AR16, AR17, AR21, AR22, AT18, AT19, AT20, AT23, AT24, AU18, AU19, AU20, AU23, AU24				
COREVDDPST	AU27, AV27, AU28, AV28, AU35, AV35, AU36, AV36, AB42, AC42	Напряжение питания периферии КП подсистем, принадлежащих к домену питания PD_CORE.	SERV_VDDPST, SERV_VDDA_OTP	1.8	
LSP1_VDDO	AV16, AV17	Напряжение питания периферии КП подсистемы lsp1_subs		1.8, 3.3 (по сбросу – 3.3)[1]	
HSP_VDDO_NFC	AL7, AL8, AM7, AM8	Напряжение питания периферии КП выводов домена Nand flash контроллера.		1.8, 3.3 (по сбросу – 3.3)[1]	

[1] Все КП домена по сбросу настроены на работу от номинала в 3.3 В.

Название выводов	Вывод на корпусе	Назначение	Объединяемые в корпусе линии питания	Номиналы напряжений, В	Число выводов
HSP_VDDO_SDMMC0	BA3, BA4	Напряжение питания периферии КП выводов домена SDMMC0		1.8, 3.3 (по сбросу – 3.3)[1]	
HSP_VDDO_SDMMC1	AY3, AY4	Напряжение питания периферии КП выводов домена SDMMC1		1.8, 3.3 (по сбросу – 3.3)[1]	
HSP_VDDO_MISC	AU5, AV5	Напряжение питания периферии КП выводов служебного домена MISC (служебные выводы SDMMC0, SDMMC1, USB).		1.8, 3.3 (по сбросу – 3.3)[1]	
HSP_VDDO_QSPI	AR8	Напряжение питания периферии КП выводов домена QSPI		1.8, 3.3 (по сбросу – 3.3)[1]	
HSP_VDDO_EMAC	AU7, AU8, AV7, AV8	Напряжение питания периферии КП выводов домена EMAC		1.8, 3.3 (по сбросу – 3.3)[1]	
HSP_USB0_DVDD	AU12	Низкоуровневое аналоговое и цифровое напряжение питания в режиме high-speed USB0 PHY		0.9	
HSP_USB0_VP	AU13	Низкоуровневое аналоговое и цифровое напряжение питания в режиме Super-speed USB0 PHY	HSP_USB0_VP, HSP_USB0_VPTX0	0.9	
HSP_USB0_VDDH	AV12, AV13	Высокоуровневое напряжение питания для режима Super-speed USB0 PHY	HSP_USB0_VDD330, HSP_USB0_VDDH0, HSP_USB0_VPH	3.3	
HSP_USB1_DVDD	AU10	Низкоуровневое аналоговое и цифровое напряжение питания в режиме high-speed USB1 PHY		0.9	
HSP_USB1_VP	AU11	Низкоуровневое аналоговое и цифровое напряжение питания в режиме Super-speed USB1 PHY	HSP_USB1_VP, HSP_USB1_VPTX0	0.9	
HSP_USB1_VDDH	AV10, AV11	Высокоуровневое напряжение питания для режима Super-speed USB0 PHY	HSP_USB1_VDD330, HSP_USB1_VDDH0, HSP_USB1_VPH	3.3	
LSP0_VDDA_PLL	AY32	Выделенное аналоговое напряжение питания PLL подсистемы lsp0_subs.		0.9	
LSP0_VSSA_PLL	AW32	Выделенная аналоговая земля PLL подсистемы lsp0_subs.		0	
SERV_VDDA_PLL	AW20	Выделенное аналоговое напряжение питания PLL подсистемы serv_subs.		0.9	

Название выводов	Вывод на корпусе	Назначение	Объединяемые в корпусе линии питания	Номиналы напряжений, В	Число выводов
SERV_VSSA_PLL	BA20	Выделенная аналоговая земля PLL подсистемы serv_subs.		0	
INTERC_VDDA_PLL	AV20	Выделенное аналоговое напряжение питания PLL подсистемы interconnect_subs.		0.9	
INTERC_VSSA_PLL	AY20	Выделенная аналоговая земля PLL подсистемы interconnect_subs.		0	
LSP1_VDDA_PLL	AW21	Выделенное аналоговое напряжение питания PLL подсистемы lsp1_subs.		0.9	
LSP1_VSSA_PLL	AV21	Выделенная аналоговая земля PLL подсистемы lsp1_subs.		0	
HSP_VDDA_PLL	AU6	Выделенное аналоговое напряжение питания PLL подсистемы hsperiph_subs_pads_fl.		0.9	
HSP_VSSA_PLL	AV6	Выделенная аналоговая земля PLL подсистемы hsperiph_subs_pads_fl.		0	
DDR_VDDA_PLL0	AB44	Выделенное аналоговое напряжение питания PLL0 подсистемы ddr_subs		0.9	
DDR_VSSA_PLL0	AB43	Выделенная аналоговая земля PLL0 подсистемы ddr_subs		0	
DDR_VDDA_PLL1	AC44	Выделенное аналоговое напряжение питания PLL1 подсистемы ddr_subs		0.9	
DDR_VSSA_PLL1	AC43	Выделенная аналоговая земля PLL1 подсистемы ddr_subs		0	
SERV_VDDA_TS	AY19	Выделенное аналоговое напряжение питания Temperature Sensor подсистемы serv_subs		1.8	
SERV_VDDA_VM	BA19	Выделенное аналоговое напряжение питания Voltage Monitor подсистемы serv_subs.		1.8	
DDR0_VDDQ	AA38, AA39, AA40, F38, F39, F40, G38, G39, G40, K38, K39, K40, L38, L39, L40, P38, P39, R38, R39, T38, T39, W38, W39, W40, Y38, Y39	Напряжение питания КП PHY DDR0		1.2, 1.5	
DDR0_VAA	P40	Напряжение питания PLL PHY DDR0		1.8	

Название выводов	Вывод на корпусе	Назначение	Объединяемые в корпусе линии питания	Номиналы напряжений, В	Число выводов
DDR1_VDDQ	AD38, AD39, AD40, AE38, AE39, AF38, AF39, AF40, AJ38, AJ39, AK38, AK39, AL38, AL39, AP38, AP39, AP40, AR38, AR39, AV38, AV39, AV40, AW38, AW39, AW40	Напряжение питания КП PHY DDR1		1.2, 1.5	
DDR1_VAA	AT40	Напряжение питания PLL PHY DDR1		1.8	
VPP	BC29	Напряжение программирования OTP		5.5	
VQQ	BA29	Напряжение предзаряда OTP		0.5*VPP	

Таблица 51.25. Питание домена BAT

Название выводов	Вывод на корпусе	Назначение	Объединяемые линии питания	Номиналы напряжений, В	Число выводов
BVDD	AU31, AU32, AV31, AV32	Напряжение питания домена питания PD_BAT		0.9	
BAT_VDDPST	BA30	Напряжение питания периферии CMOS КП подсистемы bat_domain		1.8	
BAT_VDDO	AW28	Напряжение питания периферии LVDS КП подсистемы bat_domain		1.8	

Таблица 51.26. Питание домена CPU

Название выводов	Вывод на корпусе	Назначение	Объединяемые линии питания	Номиналы напряжений, В	Число выводов
AVDD	AJ25, AJ26 AJ27, AJ28, AJ29, AK25, AK26, AK27, AK30, AK31, AK34, AK35, AL25, AL26, AL27, AL30, AL31, AL34, AL35, AM28, AM29,	Напряжение питания домена питания PD_CPU		0.9	

Название выводов	Вывод на корпусе	Назначение	Объединяемые линии питания	Номиналы напряжений, В	Число выводов
	AM32, AM33, AN28, AN29, AN32, AN33, AP25, AP26, AP27, AP28, AP29, AP32, AP33, AR25, AR26, AR27, AR30, AR31, AR34, AR35, AT25, AT26, AT27, AT30, AT31, AT34, AT35				
CPU_VDDPST	AW30	Напряжение питания периферии CMOS КП подсистемы cpu_subs		1.8	
CPU_VDDA_PLL	BB31	Выделенное аналоговое напряжение питания PLL подсистемы cpu_subs.		0.9	
CPU_VSSA_PLL	BA31	Выделенная аналоговая земля PLL подсистемы cpu_subs.		0	
CPU_VDDA_TS	AW31	Выделенное аналоговое напряжение питания Temperature Sensor подсистемы cpu_subs.		1.8	
CPU_VDDA_VM	AY31	Выделенное аналоговое напряжение питания Voltage Monitor подсистемы cpu_subs		1.8	

Таблица 51.27. Питание домена MEDIA

Название выводов	Вывод на корпусе	Назначение	Объединяемые линии питания	Номиналы напряжений, В	Число выводов
MVDD	AA26, AA27, AA28, AB23, AB24, AB25, AB28, AB29, AC23, AC24, AC25, AC28, AC29, AD25, AD26, AD27, AD30, AD31, AE26, AE27, AE30, AE31, AF26, AF27, AF30, AF31, G25, H25, H26, H27, H31, H32, H33, H36, H37, J25, J26,	Напряжение питания домена MEDIA		0.9	

Название выводов	Вывод на корпусе	Назначение	Объединяемые линии питания	Номиналы напряжений, В	Число выводов
	J27, J31, J32, J33, J36, J37, K23, K24, K25, K29, K30, K31, K34, K35, L23, L24, L25, L29, L30, L31, L34, L35, M26, M27, M28 N26, N27, N28, P23, P24, P25, P29, P30, P31 R23, R24, R25, R29, R30, R31, T26, T27, T28, U26, U27, U28, V23, V24, V25, V29, V30, V31, W23, W24, W25, W29, W30, W31, Y26, Y27, Y28, Y31				
MEDIA_VDDPST	E35, E36, F35, F36, G35, G36	Напряжение питания периферии CMOS КП подсистемы media_subs		1.8	
MEDIA_VDDA_PLL0	A35	Выделенное аналоговое напряжение питания PLL0 подсистемы media_subs.		0.9	
MEDIA_VSSA_PLL0	B35	Выделенная аналоговая земля PLL0 подсистемы media_subs.		0	
MEDIA_VDDA_PLL1	A36	Выделенное аналоговое напряжение питания PLL1 подсистемы media_subs		0.9	
MEDIA_VSSA_PLL1	B36	Выделенная аналоговая земля PLL1 подсистемы media_subs		0	
MEDIA_VDDA_PLL2	C35	Выделенное аналоговое напряжение питания PLL2 подсистемы media_subs		0.9	
MEDIA_VSSA_PLL2	D35	Выделенная аналоговая земля PLL2 подсистемы media_subs		0	
MEDIA_VDDA_PLL3	C36	Выделенное аналоговое напряжение питания PLL3 подсистемы media_subs.		0.9	
MEDIA_VSSA_PLL3	D36	Выделенная аналоговая земля PLL3 подсистемы media_subs.		0	
MEDIA_VDDA_TS	F19	Выделенное аналоговое напряжение питания Temperature Sensor подсистемы media_subs		1.8	

Название выводов	Вывод на корпусе	Назначение	Объединяемые линии питания	Номиналы напряжений, В	Число выводов
MEDIA_VDDA_VM	E19	Выделенное аналоговое напряжение питания Voltage Monitor подсистемы media_subs		1.8	
MEDIA_MIPI_TX_VP	G28	Аналоговое напряжение питания MIPI_TX	MEDIA_MIPI_TX_VPL, MEDIA_MIPI_TX_VP	0.9	
MEDIA_MIPI_TX_VPH	G26	Аналоговое напряжение питания MIPI_TX		1.8	
MEDIA_MIPI_RX0_VP	G30	Аналоговое напряжение питания MIPI_RX0		0.9	
MEDIA_MIPI_RX1_VP	G34	Аналоговое напряжение питания MIPI_RX1		0.9	
MEDIA_MIPI_RX_VPH	G31, G32	Аналоговое напряжение питания MIPI_RX0, MIPI_RX1	MEDIA_MIPI_RX0_VPH, MEDIA_MIPI_RX1_VPH	1.8	
MEDIA_TAVDD	E37	Аналоговое напряжение питания аналоговых КП TS_VCAL, TS_VSENSE подсистемы media_subs		1.8	
MEDIA_TAVSS	F37	Аналоговая земля аналоговых КП TS_VCAL, TS_VSENSE подсистемы media_subs		0	

Таблица 51.28. Питание домена SDR

Название выводов	Вывод на корпусе	Назначение	Объединяемые линии питания	Номиналы напряжений, В	Число выводов
SVDD	AA10, AA14, AA15, AA16, AA20, AA21, AA22, AA9, AB11, AB12, AB13, AB17, AB18, AB19, AC11, AC12, AC13, AC17, AC18, AC19, AC8, AD10, AD14, AD15, AD16, AD20, AD21, AD22, AD8, AD9, AE10, AE14, AE15, AE16, AE20, AE21, AE22, AE8, AE9, AF11, AF12, AF13, AF17, AF18, AF19, AF9, AG11, AG12,	Напряжение питания домена SDR		0.9	

Название выводов	Вывод на корпусе	Назначение	Объединяемые линии питания	Номиналы напряжений, В	Число выводов
	AG13, AG17, AG18, AG19, AH14, AH15, AH16, AH20, AH21, AH22, AJ14, AJ15, AJ16, AJ20, AJ21, AJ22, AK17, AK18, AK19, AL17, AL18, AL19, AM14, AM15, AM16, AM19, AM20, AN14, AN15, AN16, AN19, AN20, G21, G22, H21, H22, J11, J12, J15, J16, J19, J20, K11, K12, K15, K16, K19, K20, L11, L12, L15, L16, L19, L20, M10, M13, M14, M17, M18, M21, M22, M8, M9, N10, N13, N14, N17, N18, N21, N22, N8, N9, P11, P12, P15, P16, P19, P20, R11, R12, R15, R16, R19, R20, T10, T14, T15, T16, T20, T21, T22, T9, U10, U14, U15, U16, U20, U21, U22, U9, V11,V12, V13, V17, V18, V19, w11, W12, W13, W17, W18, W19, Y10, Y14, Y15, Y16, Y20, Y21, Y22, Y9				
SDR_VDDPST	G7, H7, T7, T8, U7, U8, Y7, Y8, AA7, AA8	Напряжение питания периферии CMOS КП подсистемы sdr_subs.		1.8	

Название выводов	Вывод на корпусе	Назначение	Объединяемые линии питания	Номиналы напряжений, В	Число выводов
SDR_VDDO_T_RX	F5	Напряжение питания периферии LVDS RX КП, находящихся в верхней части флорплана подсистемы sdr_subs.		1.8	
SDR_VDDO_R_RX	J6, J7, K6, K7, L6, L7	Напряжение питания периферии LVDS RX КП, находящихся в правой части флорплана подсистемы sdr_subs.		1.8	
SDR_VDDO_TX	AE6, AE7, AF6, AF7, AG6, AG7	Напряжение питания периферии LVDS TX КП подсистемы sdr_subs.		1.8	
SDR_PCIE0_VPH	H13	Аналоговое напряжение питания для PCI0 PHY		1.8, 1.5	
SDR_PCIE0_VP	G10, G11, H10, H11	Аналоговое напряжение питания для PCI0 PHY	SDR_PCIE0_VP SDR_PCIE0_VPTX0 SDR_PCIE0_VPTX1 SDR_PCIE0_VPTX2 SDR_PCIE0_VPTX3 SDR_PCIE0_VPDIG	0.9	
SDR_PCIE1_VPH	H15	Аналоговое напряжение питания для PCI1 PHY		1.8, 1.5	
SDR_PCIE1_VP	G17, G18, H17, H18	Аналоговое напряжение питания для PCI1 PHY	SDR_PCIE1_VP SDR_PCIE1_VPTX0 SDR_PCIE1_VPTX1 SDR_PCIE1_VPTX2 SDR_PCIE1_VPTX3 SDR_PCIE1_VPDIG	0.9	
SDR_VDDA_PLL0	R6	Выделенное аналоговое напряжение питания PLL0 подсистемы sdr_subs.		0.9	
SDR_VSSA_PLL0	R5	Выделенная аналоговая земля PLL0 подсистемы sdr_subs.		0	
SDR_VDDA_PLL1	P6	Выделенное аналоговое напряжение питания PLL1 подсистемы sdr_subs.		0.9	
SDR_VSSA_PLL1	P5	Выделенная аналоговая земля PLL1 подсистемы sdr_subs.		0	
SDR_VDDA_PLL2	P3	Выделенное аналоговое напряжение питания PLL2 подсистемы sdr_subs.		0.9	
SDR_VSSA_PLL2	P4	Выделенная аналоговая земля PLL2 подсистемы sdr_subs.		0	

Название выводов	Вывод на корпусе	Назначение	Объединяемые линии питания	Номиналы напряжений, В	Число выводов
SDR_VDDA_TS	P1	Выделенное аналоговое напряжение питания Temperature Sensor подсистемы sdr_subs.		1.8	
SDR_VDDA_VM	P2	Выделенное аналоговое напряжение питания Voltage Monitor подсистемы sdr_subs.		1.8	

51.3 Расположение выводов на корпусе

Расположение выводов микросхемы в корпусе HFCBGA-1936 приведено на рисунках: Рисунок 51.1, Рисунок 51.2, Рисунок 51.3, Рисунок 51.4.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
A	A1 DFE_ADC2_CLKX	A2 DFE_ADC2_CLK	A3 JESD1_CLKX	A4 JESD0_CLKX	A5 DFE_CLK	A6 JESDO_SYSREF	A7 DFE_GPIO12	A8 DFE_GPIO13	A9 PCIO_PHY_RESREF	A10 PCIO_TXN1	A11 VSS	A12 PCIO_RXN1	A13 VSS	A14 PCIO_REF_CLK_M	A15 VSS	A16 PCI1_TXN0	A17 VSS	A18 PCI1_RXN0	A19 CMOS0_CLK	A20 VTEST3	A21 VDATA1	A22 VDATA3
B	B1 DFE_ADC2_D7X	B2 DFE_ADC2_D7	B3 JESD1_CLK	B4 JESD0_CLK	B5 DFE_CLKX	B6 JESDO_SYSREFX	B7 DFE_GPIO9	B8 DFE_GPIO8	B9 PCI1_PHY_RESREF	B10 PCIO_TXPX1	B11 PCIO_TXN0	B12 PCIO_RXPX1	B13 PCIO_RXN0	B14 PCIO_REFCLK_PX	B15 PCI1_TXN1	B16 PCI1_TXPX0	B17 PCI1_RXN1	B18 PCI1_RXPX0	B19 CMOS0_FSYNC	B20 VTEST1	B21 VDATA5	B22 VDATA9
C	C1 DFE_ADC2_D6X	C2 DFE_ADC2_D6	C3 JESD_SYNCB_O	C4 SPARE_LVDS_TX	C5 SEG3_RTPAD	C6 JESD1_SYSREF	C7 DFE_GPIO5	C8 DFE_GPIO4	C9 DFE_GPIO1	C10 VSS	C11 PCIO_TXPX0	C12 VSS	C13 PCIO_RXPX0	C14 VSS	C15 PCI1_TXPX1	C16 VSS	C17 PCI1_RXPX1	C18 VSS	C19 CMOS1_CLK	C20 VTEST7	C21 VDATA11	C22 VDATA19
D	D1 DFE_ADC2_D5X	D2 DFE_ADC2_D5	D3 JESD_SYNCB_OX	D4 SPARE_LVDS_TXX	D5 SEG3_RTPADX	D6 JESD1_SYSREFX	D7 DFE_GPIO15	D8 DFE_GPIO14	D9 DFE_GPIO0	D10 PCIO_TXN3	D11 VSS	D12 PCIO_RXN3	D13 VSS	D14 PCI1_REF_CLK_PX	D15 VSS	D16 PCI1_TXN2	D17 VSS	D18 PCI1_RXN2	D19 CMOS1_FSYNC	D20 VTEST5	D21 VDATA2	D22 VDATA6
E	E1 DFE_ADC2_D4X	E2 DFE_ADC2_D4	E3 DFE_ADC2_FRAME	E4 SPARE_LVDS_RXX	E5 SEG3_REF0	E6 DFE_TIMER	E7 DFE_GPIO11	E8 DFE_GPIO10	E9 DFE_GPIO7	E10 PCIO_TXPX3	E11 PCIO_TXN2	E12 PCIO_RXPX3	E13 PCIO_RXN2	E14 PCI1_REF_CLK_M	E15 PCI1_TXN3	E16 PCI1_TXPX2	E17 PCI1_RXN3	E18 PCI1_RXPX2	E19 MEDIA_VDDA_VM	E20 VTEST0	E21 VTEST6	E22 VDATA12
F	F1 DFE_ADC2_D3X	F2 DFE_ADC2_D3	F3 DFE_ADC2_FRAMEX	F4 SPARE_LVDS_RX	F5 SDR_VDDO_T_RX	F6 DFE_ADC4_CLK	F7 DFE_GPIO6	F8 DFE_GPIO3	F9 DFE_GPIO2	F10 VSS	F11 PCIO_TXPX2	F12 VSS	F13 PCIO_RXPX2	F14 VSS	F15 PCI1_TXPX3	F16 VSS	F17 PCI1_RXPX3	F18 VSS	F19 MEDIA_VDDA_TS	F20 VTEST2	F21 VTEST4	F22 VDATA4
G	G1 DFE_ADC2_D2X	G2 DFE_ADC2_D2	G3 DFE_ADC2_D1X	G4 DFE_ADC2_D1	G5 VSS	G6 VSS	G7 SDR_VDDPST	G8 VSS	G9 VSS	G10 SDR_PCIE0_VP	G11 SDR_PCIE0_VP	G12 PCIO_WAKE	G13 PCIO_CLKREQ	G14 PCI0_APP_LTSSM_EN	G15 PCI1_CLKREQ	G16 PCI1_WAKE	G17 SDR_PCIE1_VP	G18 SDR_PCIE1_VP	G19 VSS	G20 VSS	G21 SVDD	G22 SVDD
H	H1 DFE_ADC2_D0X	H2 DFE_ADC2_D0	H3 SEG1_RTPADX	H4 SEG1_RTPAD	H5 SEG1_REF5	H6 VSS	H7 SDR_VDDPST	H8 VSS	H9 VSS	H10 SDR_PCIE0_VP	H11 SDR_PCIE0_VP	H12 PCIO_PERSTN	H13 SDR_PCIE0_VPH	H14 PCI1_APP_LTSSM_EN	H15 SDR_PCIE1_VPH	H16 PCI1_PERSTN	H17 SDR_PCIE1_VP	H18 SDR_PCIE1_VP	H19 VSS	H20 VSS	H21 SVDD	H22 SVDD
J	J1 DFE_ADC1_FRAMEX	J2 DFE_ADC1_FRAME	J3 DFE_ADC1_CLKX	J4 DFE_ADC1_CLK	J5 SEG1_REF4	J6 SDR_VDDO_R_RX	J7 SDR_VDDO_R_RX	J8 VSS	J9 VSS	J10 VSS	J11 SVDD	J12 SVDD	J13 VSS	J14 VSS	J15 SVDD	J16 SVDD	J17 VSS	J18 VSS	J19 SVDD	J20 SVDD	J21 VSS	J22 VSS
K	K1 DFE_ADC1_D7X	K2 DFE_ADC1_D7	K3 DFE_ADC1_D6X	K4 DFE_ADC1_D6	K5 SEG1_REF3	K6 SDR_VDDO_R_RX	K7 SDR_VDDO_R_RX	K8 VSS	K9 VSS	K10 VSS	K11 SVDD	K12 SVDD	K13 VSS	K14 VSS	K15 SVDD	K16 SVDD	K17 VSS	K18 VSS	K19 SVDD	K20 SVDD	K21 VSS	K22 VSS
L	L1 DFE_ADC1_D5X	L2 DFE_ADC1_D5	L3 DFE_ADC1_D4X	L4 DFE_ADC1_D4	L5 SEG1_REF2	L6 SDR_VDDO_R_RX	L7 SDR_VDDO_R_RX	L8 VSS	L9 VSS	L10 VSS	L11 SVDD	L12 SVDD	L13 VSS	L14 VSS	L15 SVDD	L16 SVDD	L17 VSS	L18 VSS	L19 SVDD	L20 SVDD	L21 VSS	L22 VSS
M	M1 DFE_ADC1_D3X	M2 DFE_ADC1_D3	M3 DFE_ADC1_D2X	M4 DFE_ADC1_D2	M5 SEG1_REF1	M6 VSS	M7 VSS	M8 SVDD	M9 SVDD	M10 SVDD	M11 VSS	M12 VSS	M13 SVDD	M14 SVDD	M15 VSS	M16 VSS	M17 SVDD	M18 SVDD	M19 VSS	M20 VSS	M21 SVDD	M22 SVDD
N	N1 DFE_ADC1_D1X	N2 DFE_ADC1_D1	N3 DFE_ADC1_D0X	N4 DFE_ADC1_D0	N5 SEG1_REF0	N6 VSS	N7 VSS	N8 SVDD	N9 SVDD	N10 SVDD	N11 VSS	N12 VSS	N13 SVDD	N14 SVDD	N15 VSS	N16 VSS	N17 SVDD	N18 SVDD	N19 VSS	N20 VSS	N21 SVDD	N22 SVDD
P	P1 SDR_VDDA_TS	P2 SDR_VDDA_VM	P3 SDR_VDDA_PLL2	P4 SDR_VSSA_PLL2	P5 SDR_VSSA_PLL1	P6 SDR_VDDA_PLL1	P7 VSS	P8 VSS	P9 VSS	P10 VSS	P11 SVDD	P12 SVDD	P13 VSS	P14 VSS	P15 SVDD	P16 SVDD	P17 VSS	P18 VSS	P19 SVDD	P20 SVDD	P21 VSS	P22 VSS
R	R1 DFE_GPIO30	R2 DFE_GPIO31	R3 DFE_GPIO26	R4 DFE_GPIO28	R5 SDR_VSSA_PLL0	R6 SDR_VDDA_PLL0	R7 VSS	R8 VSS	R9 VSS	R10 VSS	R11 SVDD	R12 SVDD	R13 VSS	R14 VSS	R15 SVDD	R16 SVDD	R17 VSS	R18 VSS	R19 SVDD	R20 SVDD	R21 VSS	R22 VSS
T	T1 DFE_GPIO27	T2 DFE_GPIO22	T3 DFE_GPIO23	T4 DFE_GPIO29	T5 DFE_GPIO24	T6 DFE_GPIO25	T7 SDR_VDDPST	T8 SDR_VDDPST	T9 SVDD	T10 SVDD	T11 VSS	T12 VSS	T13 VSS	T14 SVDD	T15 SVDD	T16 SVDD	T17 VSS	T18 VSS	T19 VSS	T20 SVDD	T21 SVDD	T22 SVDD
U	U1 DFE_GPIO18	U2 DFE_GPIO19	U3 DFE_GPIO20	U4 DFE_GPIO21	U5 DFE_GPIO16	U6 DFE_GPIO17	U7 SDR_VDDPST	U8 SDR_VDDPST	U9 SVDD	U10 SVDD	U11 VSS	U12 VSS	U13 VSS	U14 SVDD	U15 SVDD	U16 SVDD	U17 VSS	U18 VSS	U19 VSS	U20 SVDD	U21 SVDD	U22 SVDD
V	V1 DFE_DAC3_D14	V2 DFE_DAC3_D15	V3 DFE_DAC3_D10	V4 DFE_DAC3_D12	V5 DFE_DAC3_D13	V6 DFE_DAC3_D8	V7 VSS	V8 VSS	V9 VSS	V10 VSS	V11 SVDD	V12 SVDD	V13 SVDD	V14 VSS	V15 VSS	V16 VSS	V17 SVDD	V18 SVDD	V19 SVDD	V20 VSS	V21 VSS	V22 VSS
W	W1 DFE_DAC3_D11	W2 DFE_DAC3_D6	W3 DFE_DAC3_D7	W4 DFE_DAC3_D9	W5 DFE_DAC3_D4	W6 DFE_DAC3_D5	W7 VSS	W8 VSS	W9 VSS	W10 VSS	W11 SVDD	W12 SVDD	W13 SVDD	W14 VSS	W15 VSS	W16 VSS	W17 SVDD	W18 SVDD	W19 SVDD	W20 VSS	W21 VSS	W22 VSS
Y	Y1 DFE_DAC3_D2	Y2 DFE_DAC3_D3	Y3 DFE_DAC3_D0	Y4 DFE_DAC3_D1	Y5 DFE_DAC3_CLK	Y6 DFE_PPS	Y7 SDR_VDDPST	Y8 SDR_VDDPST	Y9 SVDD	Y10 SVDD	Y11 VSS	Y12 VSS	Y13 VSS	Y14 SVDD	Y15 SVDD	Y16 SVDD	Y17 VSS	Y18 VSS	Y19 VSS	Y20 SVDD	Y21 SVDD	Y22 SVDD
AA	AA1 GNSS_D10	AA2 GNSS_D11	AA3 GNSS_D6	AA4 GNSS_D8	AA5 GNSS_D9	AA6 GNSS_D4	AA7 SDR_VDDPST	AA8 SDR_VDDPST	AA9 SVDD	AA10 SVDD	AA11 VSS	AA12 VSS	AA13 VSS	AA14 SVDD	AA15 SVDD	AA16 SVDD	AA17 VSS	AA18 VSS	AA19 VSS	AA20 SVDD	AA21 SVDD	AA22 SVDD

Рисунок 51.1. Расположение выводов на корпусе (левый верхний угол)

AB	AB1 GNSS_D7	AB2 GNSS_D2	AB3 GNSS_D3	AB4 GNSS_D5	AB5 GNSS_D0	AB6 GNSS_D1	AB7 VSS	AB8 VSS	AB9 VSS	AB10 VSS	AB11 SVDD	AB12 SVDD	AB13 SVDD	AB14 VSS	AB15 VSS	AB16 VSS	AB17 SVDD	AB18 SVDD	AB19 SVDD	AB20 VSS	AB21 VSS	AB22 VSS
AC	AC1 DFE_DAC1_CLK	AC2 DFE_DAC1_CLKX	AC3 GNSS_CLK	AC4 GNSS_PPS	AC5 VSS	AC6 VSS	AC7 VSS	AC8 SVDD	AC9 VSS	AC10 VSS	AC11 SVDD	AC12 SVDD	AC13 SVDD	AC14 VSS	AC15 VSS	AC16 VSS	AC17 SVDD	AC18 SVDD	AC19 SVDD	AC20 VSS	AC21 VSS	AC22 VSS
AD	AD1 DFE_DAC1_D14X	AD2 DFE_DAC1_D14	AD3 DFE_DAC1_D15X	AD4 DFE_DAC1_D15	AD5 VSS	AD6 VSS	AD7 VSS	AD8 SVDD	AD9 SVDD	AD10 SVDD	AD11 VSS	AD12 VSS	AD13 VSS	AD14 SVDD	AD15 SVDD	AD16 SVDD	AD17 VSS	AD18 VSS	AD19 VSS	AD20 SVDD	AD21 SVDD	AD22 SVDD
AE	AE1 DFE_DAC1_D12X	AE2 DFE_DAC1_D12	AE3 DFE_DAC1_D13X	AE4 DFE_DAC1_D13	AE5 VSS	AE6 SDR_VDDO_TX	AE7 SDR_VDDO_TX	AE8 SVDD	AE9 SVDD	AE10 SVDD	AE11 VSS	AE12 VSS	AE13 VSS	AE14 SVDD	AE15 SVDD	AE16 SVDD	AE17 VSS	AE18 VSS	AE19 VSS	AE20 SVDD	AE21 SVDD	AE22 SVDD
AF	AF1 DFE_DAC1_D10X	AF2 DFE_DAC1_D10	AF3 DFE_DAC1_D11X	AF4 DFE_DAC1_D11	AF5 SEG2_REF4	AF6 SDR_VDDO_TX	AF7 SDR_VDDO_TX	AF8 VSS	AF9 SVDD	AF10 VSS	AF11 SVDD	AF12 SVDD	AF13 SVDD	AF14 VSS	AF15 VSS	AF16 VSS	AF17 SVDD	AF18 SVDD	AF19 SVDD	AF20 VSS	AF21 VSS	AF22 VSS
AG	AG1 DFE_DAC1_D8X	AG2 DFE_DAC1_D8	AG3 DFE_DAC1_D9X	AG4 DFE_DAC1_D9	AG5 SEG2_REF3	AG6 SDR_VDDO_TX	AG7 SDR_VDDO_TX	AG8 VSS	AG9 VSS	AG10 VSS	AG11 SVDD	AG12 SVDD	AG13 SVDD	AG14 VSS	AG15 VSS	AG16 VSS	AG17 SVDD	AG18 SVDD	AG19 SVDD	AG20 VSS	AG21 VSS	AG22 VSS
AH	AH1 DFE_DAC1_D6X	AH2 DFE_DAC1_D6	AH3 DFE_DAC1_D7X	AH4 DFE_DAC1_D7	AH5 SEG2_REF2	AH6 VSS	AH7 VSS	AH8 VSS	AH9 VSS	AH10 VSS	AH11 VSS	AH12 VSS	AH13 VSS	AH14 SVDD	AH15 SVDD	AH16 SVDD	AH17 VSS	AH18 VSS	AH19 VSS	AH20 SVDD	AH21 SVDD	AH22 SVDD
AJ	AJ1 DFE_DAC1_D4X	AJ2 DFE_DAC1_D4	AJ3 DFE_DAC1_D5X	AJ4 DFE_DAC1_D5	AJ5 SEG2_REF1	AJ6 VSS	AJ7 VSS	AJ8 CVDD	AJ9 CVDD	AJ10 CVDD	AJ11 VSS	AJ12 VSS	AJ13 VSS	AJ14 SVDD	AJ15 SVDD	AJ16 SVDD	AJ17 VSS	AJ18 VSS	AJ19 VSS	AJ20 SVDD	AJ21 SVDD	AJ22 SVDD
AK	AK1 DFE_DAC1_D2X	AK2 DFE_DAC1_D2	AK3 DFE_DAC1_D3X	AK4 DFE_DAC1_D3	AK5 SEG2_REF0	AK6 VSS	AK7 VSS	AK8 CVDD	AK9 CVDD	AK10 CVDD	AK11 CVDD	AK12 CVDD	AK13 CVDD	AK14 VSS	AK15 VSS	AK16 VSS	AK17 SVDD	AK18 SVDD	AK19 SVDD	AK20 VSS	AK21 VSS	AK22 VSS
AL	AL1 DFE_DAC1_D0X	AL2 DFE_DAC1_D0	AL3 DFE_DAC1_D1X	AL4 DFE_DAC1_D1	AL5 NFC_IO15	AL6 NFC_IO14	AL7 HSP_VDDO_NFC	AL8 HSP_VDDO_NFC	AL9 VSS	AL10 VSS	AL11 CVDD	AL12 CVDD	AL13 CVDD	AL14 VSS	AL15 VSS	AL16 VSS	AL17 SVDD	AL18 SVDD	AL19 SVDD	AL20 VSS	AL21 VSS	AL22 VSS
AM	AM1 NFC_DQS	AM2 NFC_IO11	AM3 NFC_IO7	AM4 NFC_IO13	AM5 NFC_IO10	AM6 NFC_IO12	AM7 HSP_VDDO_NFC	AM8 HSP_VDDO_NFC	AM9 VSS	AM10 VSS	AM11 VSS	AM12 VSS	AM13 VSS	AM14 SVDD	AM15 SVDD	AM16 SVDD	AM17 VSS	AM18 VSS	AM19 SVDD	AM20 SVDD	AM21 VSS	AM22 VSS
AN	AN1 NFC_IO9	AN2 NFC_IO4	AN3 NFC_IO6	AN4 NFC_IO3	AN5 NFC_IO8	AN6 NFC_IO5	AN7 VSS	AN8 VSS	AN9 CVDD	AN10 CVDD	AN11 CVDD	AN12 VSS	AN13 VSS	AN14 SVDD	AN15 SVDD	AN16 SVDD	AN17 VSS	AN18 VSS	AN19 SVDD	AN20 SVDD	AN21 VSS	AN22 VSS
AP	AP1 NFC_IO0	AP2 NFC_IO2	AP3 NFC_WEN	AP4 NFC_RBN1	AP5 NFC_IO1	AP6 NFC_RBN0	AP7 VSS	AP8 VSS	AP9 VSS	AP10 CVDD	AP11 CVDD	AP12 CVDD	AP13 VSS	AP14 VSS	AP15 CVDD	AP16 CVDD	AP17 CVDD	AP18 VSS	AP19 VSS	AP20 VSS	AP21 CVDD	AP22 CVDD
AR	AR1 NFC_ALE	AR2 NFC_CEN1	AR3 NFC_REN	AR4 NFC_CLE	AR5 NFC_CEN0	AR6 QSPI1_SS0	AR7 QSPI1_SS1	AR8 HSP_VDDO_QSPI	AR9 VSS	AR10 VSS	AR11 CVDD	AR12 CVDD	AR13 VSS	AR14 VSS	AR15 CVDD	AR16 CVDD	AR17 CVDD	AR18 VSS	AR19 VSS	AR20 VSS	AR21 CVDD	AR22 CVDD
AT	AT1 QSPI1_SCLK	AT2 QSPI1_SIO0	AT3 QSPI1_SIO1	AT4 QSPI1_SIO2	AT5 QSPI1_SIO3	AT6 QSPI1_SS2	AT7 QSPI1_SS3	AT8 VSS	AT9 VSS	AT10 VSS	AT11 VSS	AT12 VSS	AT13 VSS	AT14 VSS	AT15 VSS	AT16 VSS	AT17 VSS	AT18 CVDD	AT19 CVDD	AT20 CVDD	AT21 VSS	AT22 VSS
AU	AU1 SDMMC1_PWR	AU2 SDMMC1_18EN	AU3 SDMMC1_CDN	AU4 SDMMC1_WP	AU5 HSP_VDDO_MISC	AU6 HSP_VDDA_PLL	AU7 HSP_VDDO_EMAC	AU8 HSP_VDDO_EMAC	AU9 VSS	AU10 HSP_USB1_DVDD	AU11 HSP_USB1_VP	AU12 HSP_USB0_DVDD	AU13 HSP_USB0_VP	AU14 VSS	AU15 VSS	AU16 VSS	AU17 VSS	AU18 CVDD	AU19 CVDD	AU20 CVDD	AU21 VSS	AU22 VSS
AV	AV1 SDMMC1_DAT6	AV2 SDMMC1_DAT3	AV3 SDMMC1_DAT5	AV4 SDMMC1_DAT7	AV5 HSP_VDDO_MISC	AV6 HSP_VSSA_PLL	AV7 HSP_VDDO_EMAC	AV8 HSP_VDDO_EMAC	AV9 VSS	AV10 HSP_USB1_VDDH	AV11 HSP_USB1_VDDH	AV12 HSP_USB0_VDDH	AV13 HSP_USB0_VDDH	AV14 VSS	AV15 VSS	AV16 LSP1_VDDO	AV17 LSP1_VDDO	AV18 GPIO1_PORTD_6	AV19 GPIO1_PORTD_4	AV20 INTERC_VDDA_PLL	AV21 LSP1_VSSA_PLL	AV22 QSPIO_SS2
AW	AW1 SDMMC1_DAT2	AW2 SDMMC1_DAT4	AW3 SDMMC1_DAT1	AW4 SDMMC1_DAT0	AW5 VSS	AW6 EMAC1_RGMII_RXCTL	AW7 EMAC1_RGMII_TXCTL	AW8 EMAC0_RGMII_RXCTL	AW9 EMAC0_RGMII_TXCTL	AW10 VSS	AW11 USB1_EN_OCN	AW12 USB0_EN_OCN	AW13 VSS	AW14 USB1_RESREF	AW15 USB0_RESREF	AW16 GPIO1_PORTD_1	AW17 GPIO1_PORTD_3	AW18 GPIO1_PORTC_5	AW19 GPIO1_PORTC_7	AW20 SERV_VDDA_PLL	AW21 LSP1_VDDA_PLL	AW22 QSPIO_SS3
AY	AY1 SDMMC1_CMD	AY2 SDMMC1_CLK	AY3 HSP_VDDO_SDMMC1	AY4 HSP_VDDO_SDMMC1	AY5 EMAC1_RGMII_MDC	AY6 EMAC1_RGMII_RXC	AY7 EMAC1_RGMII_TXC	AY8 EMAC0_RGMII_RXC	AY9 EMAC0_RGMII_TXC	AY10 USB1_REF_CLK_M	AY11 USB1_ID0	AY12 USB0_REF_CLK_M	AY13 USB0_ID0	AY14 USB1_VBUS0	AY15 USB0_VBUS0	AY16 GPIO1_PORTC_2	AY17 GPIO1_PORTC_4	AY18 GPIO1_PORTB_6	AY19 SERV_VDDA_TS	AY20 INTERC_VSSA_PLL	AY21 MFBSP1_LACK	AY22 MFBSP1_LDAT7
BA	BA1 SDMMC0_PWR	BA2 SDMMC0_18EN	BA3 HSP_VDDO_SDMMC0	BA4 HSP_VDDO_SDMMC0	BA5 EMAC1_RGMII_MDIO	BA6 EMAC1_RGMII_RXD3	BA7 EMAC1_RGMII_TXD3	BA8 EMAC0_RGMII_RXD3	BA9 EMAC0_RGMII_TXD3	BA10 USB1_REF_CLK_P	BA11 USB1_TX0_M	BA12 USB0_REF_CLK_P	BA13 USB0_TX0_M	BA14 GPIO1_PORTC_0	BA15 GPIO1_PORTB_3	BA16 GPIO1_PORTB_5	BA17 GPIO1_PORTA_7	BA18 GPIO1_PORTB_1	BA19 SERV_VDDA_VM	BA20 SERV_VSSA_PLL	BA21 MFBSP1_LDAT5	BA22 MFBSP1_LDAT1
BB	BB1 SDMMC0_CDN	BB2 SDMMC0_WP	BB3 SDMMC0_CMD	BB4 SDMMC0_CLK	BB5 EMAC0_RGMII_MDC	BB6 EMAC1_RGMII_RXD2	BB7 EMAC1_RGMII_TXD2	BB8 EMAC0_RGMII_RXD2	BB9 EMAC0_RGMII_TXD2	BB10 VSS	BB11 USB1_TX0_P	BB12 VSS	BB13 USB0_TX0_P	BB14 GPIO1_PORTA_4	BB15 GPIO1_PORTA_6	BB16 GPIO1_PORTA_0	BB17 GPIO1_PORTA_2	BB18 GPIO1_PORTB_0	BB19 GPIO1_PORTA_1	BB20 NRST_WRM	BB21 MFBSP1_LDAT3	BB22 MFBSPO_LACK
BC	BC1 SDMMC0_DAT7	BC2 SDMMC0_DAT3	BC3 SDMMC0_DAT5	BC4 SDMMC0_DAT0	BC5 EMAC0_RGMII_MDIO	BC6 EMAC1_RGMII_RXD1	BC7 EMAC1_RGMII_TXD1	BC8 EMAC0_RGMII_RXD1	BC9 EMAC0_RGMII_TXD1	BC10 USB1_DM0	BC11 USB1_RX0_M	BC12 USB0_DM0	BC13 USB0_RX0_M	BC14 GPIO1_PORTD_5	BC15 GPIO1_PORTD_2	BC16 GPIO1_PORTD_0	BC17 GPIO1_PORTC_1	BC18 GPIO1_PORTB_4	BC19 GPIO1_PORTA_3	BC20 XTI274_XOUT	BC21 MFBSP1_LDAT6	BC22 MFBSP1_LDAT2
BD	BD1 SDMMC0_DAT6	BD2 SDMMC0_DAT1	BD3 SDMMC0_DAT2	BD4 SDMMC0_DAT4	BD5 CLK125	BD6 EMAC1_RGMII_RXD0	BD7 EMAC1_RGMII_TXD0	BD8 EMAC0_RGMII_RXD0	BD9 EMAC0_RGMII_TXD0	BD10 USB1_DP0	BD11 USB1_RX0_P	BD12 USB0_DP0	BD13 USB0_RX0_P	BD14 GPIO1_PORTD_7	BD15 GPIO1_PORTC_6	BD16 GPIO1_PORTC_3	BD17 GPIO1_PORTB_7	BD18 GPIO1_PORTB_2	BD19 GPIO1_PORTA_5	BD20 XTI274_XIN	BD21 MFBSP1_LCLK	BD22 MFBSP1_LDAT4

Рисунок 51.2. Расположение выводов на корпусе (левый нижний угол)

23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44
A23 VDATA7	A24 VDATA15	A25 HSYNC	A26 MIPI_TX_ DATAN2	A27 VSS	A28 MIPI_TX_ DATAN0	A29 VSS	A30 MIPI_RX0 _DATAN1	A31 VSS	A32 MIPI_RX1 _DATAN0	A33 VSS	A34 MIPI_RX1 _DATAN2	A35 MEDIA_V DDA_PLL 0	A36 MEDIA_V DDA_PLL 1	A37 VSS	A38 DDR0_D4 2	A39 DDR0_D3 7	A40 DDR0_D4 4	A41 DDR0_D4 5	A42 DDR0_D3 8	A43 DDR0_D3 9	A44 DDR0_D3 1
B23 VDATA13	B24 VDATA23	B25 DE	B26 MIPI_TX_ DATAP2	B27 MIPI_TX_ DATAN1	B28 MIPI_TX_ DATAP0	B29 MIPI_RX0 _DATAN0	B30 MIPI_RX0 _DATAP1	B31 MIPI_RX0 _DATAN2	B32 MIPI_RX1 _DATAP0	B33 MIPI_RX1 _DATAN1	B34 MIPI_RX1 _DATAP2	B35 MEDIA_V SSA_PLL0	B36 MEDIA_V SSA_PLL1	B37 VSS	B38 DDR0_D4 3	B39 VSS	B40 DDR0_D4 7	B41 DDR0_D4 6	B42 VSS	B43 DDR0_D4 0	B44 DDR0_D2 4
C23 VDATA17	C24 VDATA0	C25 VSYNC	C26 VSS	C27 MIPI_TX_ DATAP1	C28 VSS	C29 MIPI_RX0 _DATAP0	C30 VSS	C31 MIPI_RX0 _DATAP2	C32 VSS	C33 MIPI_RX1 _DATAP1	C34 VSS	C35 MEDIA_V DDA_PLL 2	C36 MEDIA_V DDA_PLL 3	C37 TS_VCAL	C38 DDR0_A3 9	C39 DDR0_D3 6	C40 DDR0_A3 5	C41 DDR0_A3 4	C42 DDR0_D4 1	C43 DDR0_D3 0	C44 DDR0_D2 5
D23 VDATA8	D24 VDATA10	D25 PIXCLK	D26 MIPI_TX_ DATAP3	D27 VSS	D28 MIPI_TX_ CLKP	D29 VSS	D30 MIPI_RX0 _REXT	D31 VSS	D32 MIPI_RX1 _CLKN	D33 VSS	D34 MIPI_RX1 _DATAN3	D35 MEDIA_V SSA_PLL2	D36 MEDIA_V SSA_PLL3	D37 TS_VSENS E	D38 DDR0_A3 8	D39 DDR0_A3 7	D40 DDR0_A3 6	D41 DDR0_A3 3	D42 DDR0_D3 5	D43 VSS	D44 DDR0_D3 2
E23 VDATA16	E24 VDATA18	E25 VDATA21	E26 MIPI_TX_ DATAN3	E27 MIPI_TX_ REXT	E28 MIPI_TX_ CLKN	E29 MIPI_RX0 _CLKN	E30 MIPI_RX0 _ATB	E31 MIPI_RX0 _DATAN3	E32 MIPI_RX1 _CLKP	E33 MIPI_RX1 _REXT	E34 MIPI_RX1 _DATAP3	E35 MEDIA_V DDPST	E36 MEDIA_V DDPST	E37 MEDIA_T AVDD	E38 DDR0_ZN	E39 DDR0_M EMRESET _L	E40 DDR0_AL ERT_N	E41 DDR0_A3 1	E42 DDR0_A3 2	E43 DDR0_D3 4	E44 DDR0_D3 3
F23 VDATA14	F24 VDATA20	F25 VDATA22	F26 VSS	F27 MIPI_TX_ ATB	F28 VSS	F29 MIPI_RX0 _CLKP	F30 VSS	F31 MIPI_RX0 _DATAP3	F32 VSS	F33 MIPI_RX1 _ATB	F34 VSS	F35 MEDIA_V DDPST	F36 MEDIA_V DDPST	F37 MEDIA_T AVSS	F38 DDR0_VD DQ	F39 DDR0_VD DQ	F40 DDR0_VD DQ	F41 DDR0_A3 0	F42 DDR0_D2 9	F43 VSS	F44 DDR0_D2 6
G23 VSS	G24 VSS	G25 MVDD	G26 MEDIA_ MIPI_TX_ VPH	G27 VSS	G28 MEDIA_ MIPI_TX_ VP	G29 VSS	G30 MEDIA_ MIPI_RX0 _VP	G31 MEDIA_ MIPI_RX_ VPH	G32 MEDIA_ MIPI_RX_ VPH	G33 VSS	G34 MEDIA_ MIPI_RX1 _VP	G35 MEDIA_V DDPST	G36 MEDIA_V DDPST	G37 VSS	G38 DDR0_VD DQ	G39 DDR0_VD DQ	G40 DDR0_VD DQ	G41 DDR0_A2 8	G42 DDR0_A2 9	G43 DDR0_D2 8	G44 DDR0_D2 7
H23 VSS	H24 VSS	H25 MVDD	H26 MVDD	H27 MVDD	H28 VSS	H29 VSS	H30 VSS	H31 MVDD	H32 MVDD	H33 MVDD	H34 VSS	H35 VSS	H36 MVDD	H37 MVDD	H38 VSS	H39 VSS	H40 DDR0_A2 7	H41 DDR0_A2 5	H42 DDR0_A2 4	H43 VSS	H44 DDR0_D1 6
J23 VSS	J24 VSS	J25 MVDD	J26 MVDD	J27 MVDD	J28 VSS	J29 VSS	J30 VSS	J31 MVDD	J32 MVDD	J33 MVDD	J34 VSS	J35 VSS	J36 MVDD	J37 MVDD	J38 VSS	J39 VSS	J40 DDR0_A2 6	J41 DDR0_A2 3	J42 DDR0_D1 8	J43 DDR0_D1 9	J44 DDR0_D1 7
K23 MVDD	K24 MVDD	K25 MVDD	K26 VSS	K27 VSS	K28 VSS	K29 MVDD	K30 MVDD	K31 MVDD	K32 VSS	K33 VSS	K34 MVDD	K35 MVDD	K36 VSS	K37 VSS	K38 DDR0_VD DQ	K39 DDR0_VD DQ	K40 DDR0_VD DQ	K41 DDR0_A2 0	K42 DDR0_A2 1	K43 VSS	K44 DDR0_D2 2
L23 MVDD	L24 MVDD	L25 MVDD	L26 VSS	L27 VSS	L28 VSS	L29 MVDD	L30 MVDD	L31 MVDD	L32 VSS	L33 VSS	L34 MVDD	L35 MVDD	L36 VSS	L37 VSS	L38 DDR0_VD DQ	L39 DDR0_VD DQ	L40 DDR0_VD DQ	L41 DDR0_A2 2	L42 DDR0_A1 9	L43 VSS	L44 DDR0_D2 1
M23 VSS	M24 VSS	M25 VSS	M26 MVDD	M27 MVDD	M28 MVDD	M29 VSS	M30 VSS	M31 VSS	M32 CVDD	M33 CVDD	M34 VSS	M35 VSS	M36 CVDD	M37 CVDD	M38 VSS	M39 VSS	M40 VSS	M41 DDR0_A1 6	M42 DDR0_A1 7	M43 DDR0_D2 3	M44 DDR0_D2 0
N23 VSS	N24 VSS	N25 VSS	N26 MVDD	N27 MVDD	N28 MVDD	N29 VSS	N30 VSS	N31 VSS	N32 CVDD	N33 CVDD	N34 VSS	N35 VSS	N36 CVDD	N37 CVDD	N38 VSS	N39 VSS	N40 VSS	N41 DDR0_A1 8	N42 DDR0_D1 2	N43 VSS	N44 DDR0_D1 3
P23 MVDD	P24 MVDD	P25 MVDD	P26 VSS	P27 VSS	P28 VSS	P29 MVDD	P30 MVDD	P31 MVDD	P32 VSS	P33 VSS	P34 CVDD	P35 CVDD	P36 VSS	P37 VSS	P38 DDR0_VD DQ	P39 DDR0_VD DQ	P40 DDR0_VA A	P41 DDR0_A1 4	P42 DDR0_A1 5	P43 DDR0_D1 5	P44 DDR0_D1 4
R23 MVDD	R24 MVDD	R25 MVDD	R26 VSS	R27 VSS	R28 VSS	R29 MVDD	R30 MVDD	R31 MVDD	R32 VSS	R33 VSS	R34 CVDD	R35 CVDD	R36 VSS	R37 VSS	R38 DDR0_VD DQ	R39 DDR0_VD DQ	R40 DDR0_VR EF	R41 DDR0_A1 2	R42 DDR0_A1 3	R43 VSS	R44 DDR0_D6
T23 VSS	T24 VSS	T25 VSS	T26 MVDD	T27 MVDD	T28 MVDD	T29 VSS	T30 VSS	T31 VSS	T32 CVDD	T33 CVDD	T34 VSS	T35 VSS	T36 CVDD	T37 CVDD	T38 DDR0_VD DQ	T39 DDR0_VD DQ	T40 DDR0_A1 0	T41 DDR0_A1 1	T42 DDR0_D5	T43 DDR0_D4	T44 DDR0_D7
U23 VSS	U24 VSS	U25 VSS	U26 MVDD	U27 MVDD	U28 MVDD	U29 VSS	U30 VSS	U31 VSS	U32 CVDD	U33 CVDD	U34 VSS	U35 VSS	U36 CVDD	U37 CVDD	U38 VSS	U39 VSS	U40 VSS	U41 DDR0_A8	U42 DDR0_A9	U43 VSS	U44 DDR0_D1 0
V23 MVDD	V24 MVDD	V25 MVDD	V26 VSS	V27 VSS	V28 VSS	V29 MVDD	V30 MVDD	V31 MVDD	V32 CVDD	V33 CVDD	V34 VSS	V35 VSS	V36 CVDD	V37 CVDD	V38 VSS	V39 VSS	V40 VSS	V41 DDR0_A6	V42 DDR0_A7	V43 VSS	V44 DDR0_D9
W23 MVDD	W24 MVDD	W25 MVDD	W26 VSS	W27 VSS	W28 VSS	W29 MVDD	W30 MVDD	W31 MVDD	W32 VSS	W33 VSS	W34 CVDD	W35 CVDD	W36 VSS	W37 VSS	W38 DDR0_VD DQ	W39 DDR0_VD DQ	W40 DDR0_VD DQ	W41 DDR0_A4	W42 DDR0_A5	W43 DDR0_D1 1	W44 DDR0_D8
Y23 VSS	Y24 VSS	Y25 VSS	Y26 MVDD	Y27 MVDD	Y28 MVDD	Y29 VSS	Y30 VSS	Y31 MVDD	Y32 VSS	Y33 VSS	Y34 CVDD	Y35 CVDD	Y36 VSS	Y37 VSS	Y38 DDR0_VD DQ	Y39 DDR0_VD DQ	Y40 DDR0_A2	Y41 DDR0_A3	Y42 DDR0_D3	Y43 VSS	Y44 DDR0_D1
AA23 VSS	AA24 VSS	AA25 VSS	AA26 MVDD	AA27 MVDD	AA28 MVDD	AA29 VSS	AA30 VSS	AA31 VSS	AA32 VSS	AA33 VSS	AA34 CVDD	AA35 CVDD	AA36 VSS	AA37 VSS	AA38 DDR0_VD DQ	AA39 DDR0_VD DQ	AA40 DDR0_VD DQ	AA41 DDR0_A0	AA42 DDR0_A1	AA43 DDR0_D2	AA44 DDR0_D0

Рисунок 51.3. Расположение выводов на корпусе (правый верхний угол)

AB23 MVDD	AB24 MVDD	AB25 MVDD	AB26 VSS	AB27 VSS	AB28 MVDD	AB29 MVDD	AB30 VSS	AB31 VSS	AB32 CVDD	AB33 CVDD	AB34 VSS	AB35 VSS	AB36 CVDD	AB37 CVDD	AB38 VSS	AB39 VSS	AB40 VSS	AB41 VSS	AB42 SERV_VD DPST	AB43 DDR_VSS A_PLL0	AB44 DDR_VDD A_PLL0
AC23 MVDD	AC24 MVDD	AC25 MVDD	AC26 VSS	AC27 VSS	AC28 MVDD	AC29 MVDD	AC30 VSS	AC31 VSS	AC32 CVDD	AC33 CVDD	AC34 VSS	AC35 VSS	AC36 CVDD	AC37 CVDD	AC38 VSS	AC39 VSS	AC40 VSS	AC41 VSS	AC42 SERV_VD DPST	AC43 DDR_VSS A_PLL1	AC44 DDR_VDD A_PLL1
AD23 VSS	AD24 VSS	AD25 MVDD	AD26 MVDD	AD27 MVDD	AD28 VSS	AD29 VSS	AD30 MVDD	AD31 MVDD	AD32 VSS	AD33 VSS	AD34 CVDD	AD35 CVDD	AD36 VSS	AD37 VSS	AD38 DDR1_VD DQ	AD39 DDR1_VD DQ	AD40 DDR1_VD DQ	AD41 DDR1_A3 8	AD42 DDR1_A3 9	AD43 DDR1_D4 1	AD44 DDR1_D4 0
AE23 VSS	AE24 VSS	AE25 VSS	AE26 MVDD	AE27 MVDD	AE28 VSS	AE29 VSS	AE30 MVDD	AE31 MVDD	AE32 VSS	AE33 VSS	AE34 CVDD	AE35 CVDD	AE36 VSS	AE37 VSS	AE38 DDR1_VD DQ	AE39 DDR1_VD DQ	AE40 DDR1_A3 6	AE41 DDR1_A3 7	AE42 DDR1_D4 2	AE43 VSS	AE44 DDR1_D4 3
AF23 VSS	AF24 VSS	AF25 VSS	AF26 MVDD	AF27 MVDD	AF28 VSS	AF29 VSS	AF30 MVDD	AF31 MVDD	AF32 VSS	AF33 VSS	AF34 CVDD	AF35 CVDD	AF36 VSS	AF37 VSS	AF38 DDR1_VD DQ	AF39 DDR1_VD DQ	AF40 DDR1_VD DQ	AF41 DDR1_A3 4	AF42 DDR1_A3 5	AF43 DDR1_D4 7	AF44 DDR1_D4 4
AG23 CVDD	AG24 CVDD	AG25 VSS	AG26 VSS	AG27 VSS	AG28 CVDD	AG29 CVDD	AG30 VSS	AG31 VSS	AG32 CVDD	AG33 CVDD	AG34 VSS	AG35 VSS	AG36 CVDD	AG37 CVDD	AG38 VSS	AG39 VSS	AG40 VSS	AG41 DDR1_A3 1	AG42 DDR1_A3 0	AG43 VSS	AG44 DDR1_D4 5
AH23 CVDD	AH24 CVDD	AH25 VSS	AH26 VSS	AH27 VSS	AH28 CVDD	AH29 CVDD	AH30 VSS	AH31 VSS	AH32 CVDD	AH33 CVDD	AH34 VSS	AH35 VSS	AH36 CVDD	AH37 CVDD	AH38 VSS	AH39 VSS	AH40 VSS	AH41 DDR1_A3 3	AH42 DDR1_A3 2	AH43 DDR1_D3 6	AH44 DDR1_D4 6
AJ23 CVDD	AJ24 CVDD	AJ25 AVDD	AJ26 AVDD	AJ27 AVDD	AJ28 AVDD	AJ29 AVDD	AJ30 VSS	AJ31 VSS	AJ32 CVDD	AJ33 CVDD	AJ34 VSS	AJ35 VSS	AJ36 CVDD	AJ37 CVDD	AJ38 DDR1_VD DQ	AJ39 DDR1_VD DQ	AJ40 DDR1_A2 8	AJ41 DDR1_A2 9	AJ42 DDR1_D3 8	AJ43 VSS	AJ44 DDR1_D3 7
AK23 VSS	AK24 VSS	AK25 AVDD	AK26 AVDD	AK27 AVDD	AK28 VSS	AK29 VSS	AK30 AVDD	AK31 AVDD	AK32 VSS	AK33 VSS	AK34 AVDD	AK35 AVDD	AK36 VSS	AK37 VSS	AK38 DDR1_VD DQ	AK39 DDR1_VD DQ	AK40 DDR1_VR EF	AK41 DDR1_A2 6	AK42 DDR1_A2 7	AK43 DDR1_D3 1	AK44 DDR1_D3 9
AL23 VSS	AL24 VSS	AL25 AVDD	AL26 AVDD	AL27 AVDD	AL28 VSS	AL29 VSS	AL30 AVDD	AL31 AVDD	AL32 VSS	AL33 VSS	AL34 AVDD	AL35 AVDD	AL36 VSS	AL37 VSS	AL38 DDR1_VD DQ	AL39 DDR1_VD DQ	AL40 DDR1_A2 5	AL41 DDR1_A2 4	AL42 DDR1_D2 4	AL43 VSS	AL44 DDR1_D3 0
AM23 CVDD	AM24 CVDD	AM25 VSS	AM26 VSS	AM27 VSS	AM28 AVDD	AM29 AVDD	AM30 VSS	AM31 VSS	AM32 AVDD	AM33 AVDD	AM34 VSS	AM35 VSS	AM36 CVDD	AM37 CVDD	AM38 VSS	AM39 VSS	AM40 VSS	AM41 DDR1_A2 2	AM42 DDR1_A2 3	AM43 DDR1_D3 5	AM44 DDR1_D3 2
AN23 CVDD	AN24 CVDD	AN25 VSS	AN26 VSS	AN27 VSS	AN28 AVDD	AN29 AVDD	AN30 VSS	AN31 VSS	AN32 AVDD	AN33 AVDD	AN34 VSS	AN35 VSS	AN36 CVDD	AN37 CVDD	AN38 VSS	AN39 VSS	AN40 VSS	AN41 DDR1_A2 1	AN42 DDR1_A2 0	AN43 VSS	AN44 DDR1_D3 3
AP23 VSS	AP24 VSS	AP25 AVDD	AP26 AVDD	AP27 AVDD	AP28 AVDD	AP29 AVDD	AP30 VSS	AP31 VSS	AP32 AVDD	AP33 AVDD	AP34 VSS	AP35 VSS	AP36 CVDD	AP37 CVDD	AP38 DDR1_VD DQ	AP39 DDR1_VD DQ	AP40 DDR1_VD DQ	AP41 DDR1_A1 8	AP42 DDR1_A1 9	AP43 DDR1_D2 9	AP44 DDR1_D3 4
AR23 VSS	AR24 VSS	AR25 AVDD	AR26 AVDD	AR27 AVDD	AR28 VSS	AR29 VSS	AR30 AVDD	AR31 AVDD	AR32 VSS	AR33 VSS	AR34 AVDD	AR35 AVDD	AR36 VSS	AR37 VSS	AR38 DDR1_VD DQ	AR39 DDR1_VD DQ	AR40 DDR1_A1 7	AR41 DDR1_A1 6	AR42 DDR1_D2 5	AR43 VSS	AR44 DDR1_D2 8
AT23 CVDD	AT24 CVDD	AT25 AVDD	AT26 AVDD	AT27 AVDD	AT28 VSS	AT29 VSS	AT30 AVDD	AT31 AVDD	AT32 VSS	AT33 VSS	AT34 AVDD	AT35 AVDD	AT36 VSS	AT37 VSS	AT38 VSS	AT39 VSS	AT40 DDR1_VA A	AT41 DDR1_A1 4	AT42 DDR1_A1 5	AT43 DDR1_D2 7	AT44 DDR1_D2 6
AU23 CVDD	AU24 CVDD	AU25 VSS	AU26 VSS	AU27 SERV_VD DPST	AU28 SERV_VD DPST	AU29 VSS	AU30 VSS	AU31 BVDD	AU32 BVDD	AU33 VSS	AU34 VSS	AU35 SERV_VD DPST	AU36 SERV_VD DPST	AU37 VSS	AU38 VSS	AU39 VSS	AU40 DDR1_A1 3	AU41 DDR1_A1 2	AU42 DDR1_D1 5	AU43 VSS	AU44 DDR1_D1 6
AV23 QSPI0_SS 1	AV24 QSPI0_SS 0	AV25 VSS	AV26 VSS	AV27 SERV_VD DPST	AV28 SERV_VD DPST	AV29 VSS	AV30 VSS	AV31 BVDD	AV32 BVDD	AV33 VSS	AV34 VSS	AV35 SERV_VD DPST	AV36 SERV_VD DPST	AV37 VSS	AV38 DDR1_VD DQ	AV39 DDR1_VD DQ	AV40 DDR1_VD DQ	AV41 DDR1_A1 0	AV42 DDR1_A1 1	AV43 DDR1_D1 4	AV44 DDR1_D1 7
AW23 QSPI0_SI O3	AW24 QSPI0_SI O1	AW25 QSPI0_SI O2	AW26 QSPI0_SI O0	AW27 QSPI0_SC LK	AW28 BAT_VDD O	AW29 BS_EN	AW30 CPU_VDD PST	AW31 CPU_VDD A_TS	AW32 LSP0_VSS A_PLL	AW33 GPIO0_P ORTD_6	AW34 GPIO0_P ORTC_7	AW35 GPIO0_P ORTB_6	AW36 GPIO0_P ORTA_7	AW37 GPIO0_P ORTA_2	AW38 DDR1_VD DQ	AW39 DDR1_VD DQ	AW40 DDR1_VD DQ	AW41 DDR1_A9	AW42 DDR1_A8	AW43 VSS	AW44 DDR1_D2 2
AY23 MFBSP0_ LDAT7	AY24 MFBSP0_ LDAT3	AY25 CPU_OFF ACKN	AY26 MEDIA_O FFREQN	AY27 TDO	AY28 TESTMOD E	AY29 VS_EN	AY30 VSS	AY31 CPU_VDD A_VM	AY32 LSP0_VD DA_PLL	AY33 GPIO0_P ORTD_2	AY34 GPIO0_P ORTC_3	AY35 GPIO0_P ORTB_7	AY36 GPIO0_P ORTB_2	AY37 GPIO0_P ORTA_3	AY38 DDR1_ZN	AY39 DDR1_M EMRESET _L	AY40 DDR1_AL ERT_N	AY41 DDR1_A6	AY42 DDR1_A7	AY43 DDR1_D2 3	AY44 DDR1_D2 1
BA23 MFBSP0_ LDAT5	BA24 MFBSP0_ LDAT1	BA25 CPU_OFF REQN	BA26 MEDIA_O FFACKN	BA27 TDI	BA28 JMODE1	BA29 VQQ	BA30 BAT_VDD PST	BA31 CPU_VSS A_PLL	BA32 GPIO0_P ORTD_7	BA33 GPIO0_P ORTD_3	BA34 GPIO0_P ORTC_6	BA35 GPIO0_P ORTC_2	BA36 GPIO0_P ORTB_3	BA37 GPIO0_P ORTA_6	BA38 DDR1_A3	BA39 DDR1_A1	BA40 DDR1_A5	BA41 DDR1_A4	BA42 DDR1_D1 3	BA43 VSS	BA44 DDR1_D2 0
BB23 MFBSP0_ LDAT6	BB24 MFBSP0_ LDAT0	BB25 I2C4_CU R_PU_EN	BB26 CLKOUT	BB27 TMS	BB28 JMODE0	BB29 EXTINT1	BB30 SERV_SPA RE1	BB31 CPU_VDD A_PLL	BB32 GPIO0_P ORTD_0	BB33 GPIO0_P ORTC_5	BB34 GPIO0_P ORTB_4	BB35 GPIO0_P ORTB_1	BB36 GPIO0_P ORTA_0	BB37 GPIO0_P ORTA_1	BB38 DDR1_A2	BB39 DDR1_D0	BB40 DDR1_A0	BB41 DDR1_D1 1	BB42 VSS	BB43 DDR1_D1 8	BB44 DDR1_D1 2
BC23 MFBSP0_ LCLK	BC24 MFBSP0_ LDAT2	BC25 I2C4_SDA	BC26 SDR_OFF ACKN	BC27 TRSTN	BC28 BOOT2	BC29 VPP	BC30 XTI32K_X OUT	BC31 CORE_OF FREQN	BC32 GPIO0_P ORTD_5	BC33 GPIO0_P ORTC_4	BC34 GPIO0_P ORTC_1	BC35 GPIO0_P ORTB_0	BC36 GPIO0_P ORTA_5	BC37 VSS	BC38 DDR1_D3	BC39 VSS	BC40 DDR1_D1 0	BC41 VSS	BC42 DDR1_D6	BC43 VSS	BC44 DDR1_D1 9
BD23 MFBSP1_ LDAT0	BD24 MFBSP0_ LDAT4	BD25 I2C4_SCL	BD26 SDR_OFF REQN	BD27 TCK	BD28 BOOT1	BD29 BOOT0	BD30 XTI32K_XI N	BD31 NRST_PO N	BD32 GPIO0_P ORTD_4	BD33 GPIO0_P ORTD_1	BD34 GPIO0_P ORTC_0	BD35 GPIO0_P ORTB_5	BD36 GPIO0_P ORTA_4	BD37 VSS	BD38 DDR1_D2	BD39 DDR1_D1	BD40 DDR1_D9	BD41 DDR1_D8	BD42 DDR1_D7	BD43 DDR1_D5	BD44 DDR1_D4

Рисунок 51.4. Расположение выводов на корпусе (правый нижний угол)

52. СПРАВОЧНАЯ ИНФОРМАЦИЯ

Информация, представленная в таблицах, носит предварительный характер и будет уточнена по результатам исследований полученных образцов микросхемы в ходе выполнения ОКР, в частности, макетных образцов.

52.1 Электрические характеристики

Таблица 52.1. Рабочие характеристики электропитания

Имя	Описание	Мин.	Ном.	Макс.
CVDD	Напряжение питания домена PD_CORE, В	0.855	0.900	0.945
BVDD	Напряжение питания домена PD_BAT, В	0.855	0.9	0.945
AVDD	Напряжение питания домена PD_CPU, В	0.855	0.9	0.945
MVDD	Напряжение питания домена PD_MEDIA, В	0.855	0.9	0.945
SVDD	Напряжение питания домена PD_SDR, В	0.855	0.9	0.945
SERV_VDDA_PLL INTERC_VDDA_PLL LSP0_VDDA_PLL LSP1_VDDA_PLL HSP_VDDA_PLL DDR_VDDA_PLL0 DDR_VDDA_PLL1	Аналоговое напряжение питания PLL подсистем, входящих в домен PD_CORE	0.855	0.9	0.945
CPU_VDDA_PLL	Аналоговое напряжение питания PLL подсистем, входящих в домен PD_CPU	0.855	0.9	0.945
MEDIA_VDDA_PLL0 MEDIA_VDDA_PLL1 MEDIA_VDDA_PLL2 MEDIA_VDDA_PLL3	Аналоговое напряжение питания PLL подсистем, входящих в домен PD_MEDIA	0.855	0.9	0.945
SDR_VDDA_PLL0 SDR_VDDA_PLL1 SDR_VDDA_PLL2	Аналоговое напряжение питания PLL подсистем, входящих в домен PD_SDR	0.855	0.9	0.945
HSP_USB0_DVDD HSP_USB0_VP HSP_USB1_DVDD HSP_USB1_VP	Аналоговые напряжения для линий питания PHY USB	0.84	0.9	0.945
MEDIA_MIPI_TX_VP MEDIA_MIPI_RX0_VP MEDIA_MIPI_RX1_VP	Аналоговые напряжения для линий питания PHY MIPI	0.84	0.9	0.945
SDR_PCIE0_VP SDR_PCIE1_VP	Аналоговые напряжения для линий питания PHY PCIE/JESD204B	0.84	0.9	0.945
COREVDDPST	Напряжение питания периферии КП подсистем, принадлежащих к домену питания PD_CORE	1.71	1.8	1.89
BAT_VDDPST	Напряжение питания периферии КП подсистем, принадлежащих к домену питания PD_BAT	1.71	1.8	1.89

CPU_VDDPST	Напряжение питания периферии КП подсистем, принадлежащих к домену питания PD_BAT	1.71	1.8	1.89
MEDIA_VDDPST	Напряжение питания периферии КП подсистем, принадлежащих к домену питания PD_MEDIA	1.71	1.8	1.89
SDR_VDDPST	Напряжение питания периферии КП подсистем, принадлежащих к домену питания PD_SDR	1.71	1.8	1.89
SDR_VDDA_TS SDR_VDDA_VM MEDIA_VDDA_TS MEDIA_VDDA_VM CPU_VDDA_TS CPU_VDDA_VM SERV_VDDA_TS SERV_VDDA_VM	Аналоговое напряжение питания встроенных датчиков температуры и напряжения	1.71	1.8	1.89
MEDIA_TAVDD	Аналоговое напряжение питания КП TS_VCAL, TS_VSENSE	1.71	1.8	1.89
MEDIA_MIPI_TX_VPH MEDIA_MIPI_RX_VPH	Аналоговое напряжение питания MIPI PHY	1.71	1.8	1.89
SDR_VDDO_T_RX SDR_VDDO_R_RX SDR_VDDO_TX	Напряжения питания периферии LVDS КП	1.71	1.8	1.89
LSP1_VDDO	Напряжение питания КП подсистемы lsperiph1	3.14 - 1.71	3.3 - 1.8	3.46 - 1.89
HSP_VDDO_NFC HSP_VDDO_SDMMC0 HSP_VDDO_SDMMC1 HSP_VDDO_MISC HSP_VDDO_QSPI HSP_VDDO_EMAC	Напряжение питания КП подсистемы hsperiph	3.14 - 1.71	3.3 - 1.8	3.46 - 1.89
BAT_VDDO	Напряжение питания КП PD_BAT	1.71	1.8	1.89
HSP_USB0_VDDH HSP_USB1_VDDH	Напряжения питания PHY USB	3.14	3.3	3.46
SDR_PCIE0_VPH SDR_PCIE1_VPH	Аналоговое напряжение питания КП PHY PCIE/JESD	1.71 - 1.425	1.8 - 1.5	1.89 - 1.575
DDR0_VAA DDR1_VAA	Аналоговое напряжение питания PLL PHY DDR	1.71	1.8	1.89
DDR0_VDDQ DDR1_VDDQ	Напряжение питания КП PHY DDR в режиме DDR3	1.425	1.5	1.575
DDR0_VDDQ DDR1_VDDQ	Напряжение питания КП PHY DDR в режиме DDR3L	1.283	1.35	1.417
DDR0_VDDQ DDR1_VDDQ	Напряжение питания КП PHY DDR в режиме DDR4	1.14	1.2	1.26
DDR0_VDDQ DDR1_VDDQ	Напряжение питания КП PHY DDR в режиме LPDDR3	1.14	1.2	1.26
DDR0_VDDQ DDR1_VDDQ	Напряжение питания КП PHY DDR в режиме LPDDR4	1.055	1.1	1.165

Таблица 52.2. Электрические характеристики резонатора 27.456 МГц (XTI274_XIN, XTI274_XOUT)

Имя	Описание	Мин.	Ном.	Макс.	Примечание
f	Фундаментальная частота, МГц	-	27.456	-	
	Стабильность частоты, ppm	50			
C_L*	Емкость Cload, пФ		8	12	
Rf	Сопротивление параллельного резистора, МОм		1		Данный резистор, рекомендуется в параллель с кристаллом осцилятора
ESR	Эквивалентное последовательное сопротивление резонатора, Ом	-	-	40	

*C_L = (C_{C1}*C_{C2})/(C_{C1}+C_{C2})+ C_{SHUNT}

Таблица 52.3. Электрические характеристики генератора 27.456 МГц (XTI274_XIN)

Имя	Описание	Мин.	Ном.	Макс.	Примечание
f	Частота, МГц	-	27.456	-	
	Стабильность частоты, ppm	50			
f_{DC}	Duty цикл, %	н.д.			
f_{JPP}	Джитер пик-пик, %	-	-	2	
t_R	Длительность переднего фронта, нс	-	-	5	
t_F	Длительность заднего фронта, нс	-	-	5	

Таблица 52.4. Электрические характеристики резонатора 32.768 кГц (XTI32K_XIN, XTI32K_XOUT)

Имя	Описание	Мин.	Ном.	Макс.	Примечание
f	Частота, КГц	-	32.768	-	
	Стабильность частоты, ppm	50			
	Стабильность duty cycle, %	7			
	Джиттер, пс	100			

Схема подключения резонатора дана в Разделе 52.4.

Таблица 52.5. Электрические характеристики корпуса микросхемы по выводам формирования частоты

Вывод	R (Ом)	L (нГн)	C (пФ)
XTO_24M	0,40874	2,826	1,482
XTI_32K	0,35916	2,531	1,353
XTO_32K	0,30975	2,217	1,288

Таблица 52.6. Электрические характеристики драйверов ввода-вывода GPIO0, QSPI0, Видео ввода/вывода, MFBSP0, MFBSP1, CMOS выводов SDR, Служебных выводов[1)]

Имя	Описание	Условия	Мин.	Ном	Макс.
V_{IH}	Входное напряжение высокого уровня, В	U_{CCP} = 1,8В±10%	0.65* U_{CCP}	-	1.98
V_{IL}	Входное напряжение низкого уровня, В	U_{CCP} = 1,8В±10%	- 0.3	-	0.35* U_{CCP}
V_{OH}	Выходное напряжение высокого уровня, В	U_{CCP} = 1,8В±10%	1.35	-	-
V_{OL}	Выходное напряжение низкого уровня, В	U_{CCP} = 1,8В±10%	-	-	0.45
I_{LKG}	Ток утечки, мкА	при V_I= 1.8В или 0В	-	-	±10

Имя	Описание	Условия	Мин.	Ном	Макс.
Rpull-up	Сопротивление резистора подтяжки, КОм		60	89	137
Rpulldown	Сопротивление резистора подтяжки, КОм		61	104	196
I_{OL}	Максимальный выходной ток низкого уровня, мА	Для Drive 4 - Для Drive 8	5.5 - 11.1	9.2 - 18.2	13.0 - 25.6
I_{OH}	Минимальный выходной ток высокого уровня, мА	Для Drive 4 - Для Drive 8	6.6 - 13.1	9.6 - 19.1	13.2 - 26.2

[1)] Кроме выводов XTI32K_XIN, XTI32K_XOUT

Таблица 52.7. Электрические характеристики драйверов ввода-вывода SDMMC0, SDMMC1, NAND, QSPI1, EMAC0, EMAC1, GPIO1

Имя	Описание	Условия	Мин.	Ном.	Макс.
V_{IH}	Входное напряжение высокого уровня, В	U_{CCP} = 3,3В±10%	2.0	-	0.3 + U_{CCP}
		U_{CCP} = 1,8В±10%	0.65 * U_{CCP}	-	
V_{IL}	Входное напряжение низкого уровня, В	U_{CCC} = 0,9В±5%, U_{CCP} = 3,3В±10%	- 0.3	-	0.8
		U_{CCC} = 0,9В±5%, U_{CCP} = 1,8В±10%		-	0.35 * U_{CCP}
V_{OH}	Выходное напряжение высокого уровня, В	I_{OH}= 4 мА, U_{CCP} = 3,3В±10% I_{OH}= 8 мА, U_{CCP} = 3,3В±10%	2.4	-	0.3 + U_{CCP}
		I_{OH}= 4 мА, U_{CCP} = 1,8В±10% I_{OH}= 8 мА, U_{CCP} = 1,8В±10%	1.4	-	0.3 + U_{CCP}
V_{OL}	Выходное напряжение низкого уровня, В	I_{OL}=4 мА, U_{CCP} = 3,3В±10% I_{OL}=8 мА, U_{CCP} = 3,3В±10%	- 0.3	-	0.4
		I_{OL}=4 мА, U_{CCP} = 1,8В±10% I_{OL}=8 мА, U_{CCP} = 1,8В±10%	- 0.3	-	0.4
I_{LKG}	Ток утечки, мкА	при V_I=3.3В			15
Rpull-up	Сопротивление резистора подтяжки, КОм	-	16	30	43
Rpulldown	Сопротивление резистора подтяжки, КОм	-	16	30	43
I_{OL}	Максимальный выходной ток низкого уровня, мА	U_{CCP} = 3,3В	8.00	-	47.61
I_{OH}	Минимальный выходной ток высокого уровня, мА	U_{CCP} = 3,3В	8.39	-	50.24

Таблица 52.8. Электрические характеристики драйверов ввода-вывода LVDS подсистемы SDR

Имя	Описание	Условия	Мин.	Ном.	Макс.
V_{OH}	Выходное напряжение высокого уровня, В	При Rline = 100 Ом	-	-	1.475
V_{OL}	Выходное напряжение низкого уровня, В	При Rline = 100 Ом	0.925	-	-
V_{OD}	Модуль выходного дифференциального напряжения, мВ	При Rline = 100 Ом	250	-	400
V_{OS}	Выходное напряжение смещения, мВ	При Rline = 100 Ом	1125	-	1275
Isa, Isb	Выходной ток, мА	При закоротке на землю	-	-	40
Iab	Выходной ток, мА	При закоротке драйверов друг на друга	-	-	12
V_I	Диапазон входного напряжения, В		0	-	1.8

Имя	Описание	Условия	Мин.	Ном.	Макс.
V_{ID}	Модуль входного дифференциального напряжения, мВ		100	-	1800
V_{HYST}	Входной дифференциальный гистерезис, мВ		25	-	-
R_I	Входной дифференциальный импеданс, Ом		80	-	120

Таблица 52.9. Фронты подачи питания

Линии питания	Наклон фронта подачи питания, максимум, мВ/мкс
CVDD, BVDD, AVDD, MVDD, SVDD	18
COREVDDPST, BAT_VDDPST, CPU_VDDPST, MEDIA_VDDPST	18
LSP_VDDO, HSP_VDDO_NFC, HSP_VDDO_SDMMC1, HSP_VDDO_SDMMC0, HSP_VDDO_MISC, HSP_VDDO_QSPI, HSP_VDDO_EMAC	120
DDR0_VAA, DDR1_VAA, DDR0_VDDQ, DDR1_VDDQ	5
MEDIA_MIPI_TX_VPH, MEDIA_MIPI_RX_VPH	100
HSP_USB0_VDDH, HSP_USB1_VDDH	5
SDR_PCIE0_VPH, SDR_PCIE1_VPH	140
Остальные	18

52.2 Энергопотребление

Таблица 52.10. Пиковые токи по цепям питания [0)]

Питание	Пиковый ток, мА
CVDD	
BVDD	
AVDD	
MVDD	
SVDD	
SERV_VDDA_PLL	
INTERC_VDDA_PLL	
LSP0_VDDA_PLL	
LSP1_VDDA_PLL	
HSP_VDDA_PLL	
DDR_VDDA_PLL0	
DDR_VDDA_PLL1	
CPU_VDDA_PLL	
MEDIA_VDDA_PLL0	
MEDIA_VDDA_PLL1	
MEDIA_VDDA_PLL2	
MEDIA_VDDA_PLL3	
SDR_VDDA_PLL0	
SDR_VDDA_PLL1	
SDR_VDDA_PLL2	
HSP_USB0_DVDD HSP_USB1_DVDD	

Питание	Пиковый ток, мА
HSP_USB0_VP HSP_USB1_VP	
MEDIA_MIPI_TX_VP	
MEDIA_MIPI_RX0_VP MEDIA_MIPI_RX1_VP	
SDR_PCIE0_VP SDR_PCIE1_VP	
COREVDDPST	
BAT_VDDPST	
CPU_VDDPST	
MEDIA_VDDPST	
SDR_VDDPST	
SERV_VDDA_TS SERV_VDDA_VM	
SDR_VDDA_TS SDR_VDDA_VM	
CPU_VDDA_TS CPU_VDDA_VM	
MEDIA_VDDA_TS MEDIA_VDDA_VM	
MEDIA_TAVDD	
MEDIA_MIPI_TX_VPH	
MEDIA_MIPI_RX_VPH	
SDR_VDDO_T_RX SDR_VDDO_R_RX	
SDR_VDDO_TX	
LSP1_VDDO HSP_VDDO_NFC HSP_VDDO_SDMMC0 HSP_VDDO_SDMMC1 HSP_VDDO_MISC HSP_VDDO_QSPI HSP_VDDO_EMAC	
BAT_VDDO	
HSP_USB0_VDDH HSP_USB1_VDDH	
SDR_PCIE0_VPH SDR_PCIE1_VPH	
DDR0_VAA DDR1_VAA	
DDR0_VDDQ DDR1_VDDQ	

0. Приведенные в таблице данные являются справочными и получены в результате измерений на единичных тестовых образцах микросхемы.

Таблица 52.11. Потребление микросхемы по ядру в различных режимах 0)

Режим	Домен питания микросхемы					Значение
	BAT	CORE	CPU	MEDIA	SDR	
Рабочий	0.9 В	0.9 В	0.9 В	0.9 В	0.9 В	
	0.9 В	0.9 В	0.9 В	0.9 В	OFF	
	0.9 В	0.9 В	0.9 В	OFF	0.9 В	
	0.9 В	0.9 В	0.9 В	OFF	OFF	
	0.9 В	0.9 В	OFF	0.9 В	0.9 В	

Режим	Домен питания микросхемы					Значение
	BAT	CORE	CPU	MEDIA	SDR	
	0.9 В	0.9 В	OFF	0.9 В	OFF	
	0.9 В	0.9 В	OFF	OFF	0.9 В	
	0.9 В	0.9 В	OFF	OFF	OFF	
OD	0.945 В	0.945 В	0.945 В	0.945 В	0.945 В	
	0.945 В	0.945 В	0.945 В	0.945 В	OFF	
	0.945 В	0.945 В	0.945 В	OFF	0.945 В	
	0.945 В	0.945 В	0.945 В	OFF	OFF	
Ожидание	0.855 В	0.855 В	OFF	OFF	OFF	
Сон	0.855 В	OFF	OFF	OFF	OFF	

0. Приведенные в таблице данные являются справочными и получены в результате измерений на единичных тестовых образцах микросхемы.

Таблица 52.12. Динамическое потребление процессорных ядер микросхемы [0)]

Напряжение питания ядра – 0.9 В	-60, С	25, С	85, С
Потребление CPU, в зависимости от задачи, на ядро, мВт/МГц			
Потребление DSP, в зависимости от задачи, на ядро, мВт/МГц			
Потребление RISC, в зависимости от задачи, на ядро, мВт/МГц			

0. Приведенные в таблице данные являются справочными и получены в результате измерений на единичных тестовых образцах микросхемы.

Таблица 52.13. Статический ток по ядру микросхемы, при напряжении 0.9 В [0)]

Температура, С	мин., мА	сред., мА	макс., мА

0. Приведенные в таблице данные являются справочными и получены в результате измерений на единичных тестовых образцах микросхемы.

Таблица 52.14. Доля отключаемых доменов питания в статическом потреблении микросхемы [0)]

Домен питания	Доля, % (в НКУ)
CORE	
CPU SYS	
CPU0	
CPU1	
CPU2	
CPU3	
MEDIA	
SDR	
BAT	

0. Приведенные в таблице данные являются справочными и получены в результате измерений на единичных тестовых образцах микросхемы.

52.3 Корпус

Механические данные корпуса – чертеж /*TODO*/

Примечание. Необходимость пассивного охлаждения корпуса микросхемы определяется разработчиком РЭА исходя из условий работы микросхемы и конструктива корпуса прибора. Отладочные платы с микросхемой имеют в комплекте опциональный радиатор пассивного охлаждения.

52.4 Подключение резонатора 32 КГц

Рекомендуемая схема подключения резонатора/кристалла 32 КГц показана на Рисунок 52.1

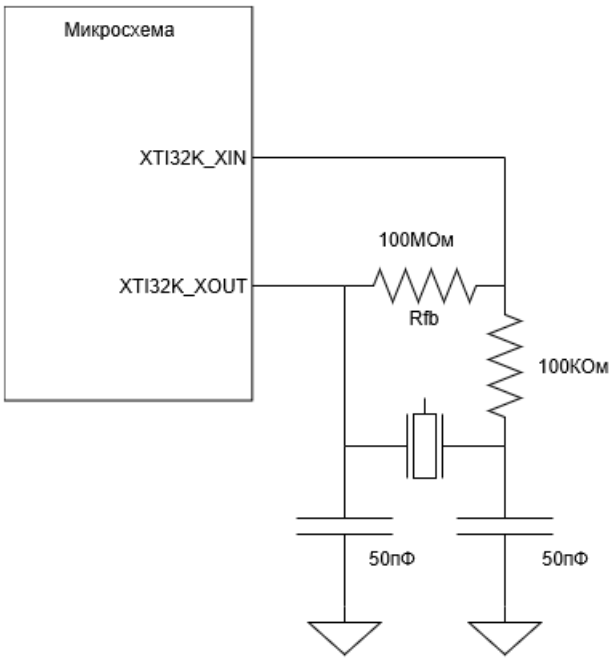


Рисунок 52.1. Рекомендуемая схема подключения резонатора 32КГц

52.5 Пример подключения блоков DAC/ADC с Jesd204b интерфейсом

Подключение к 0-му контроллеру Jesd204b

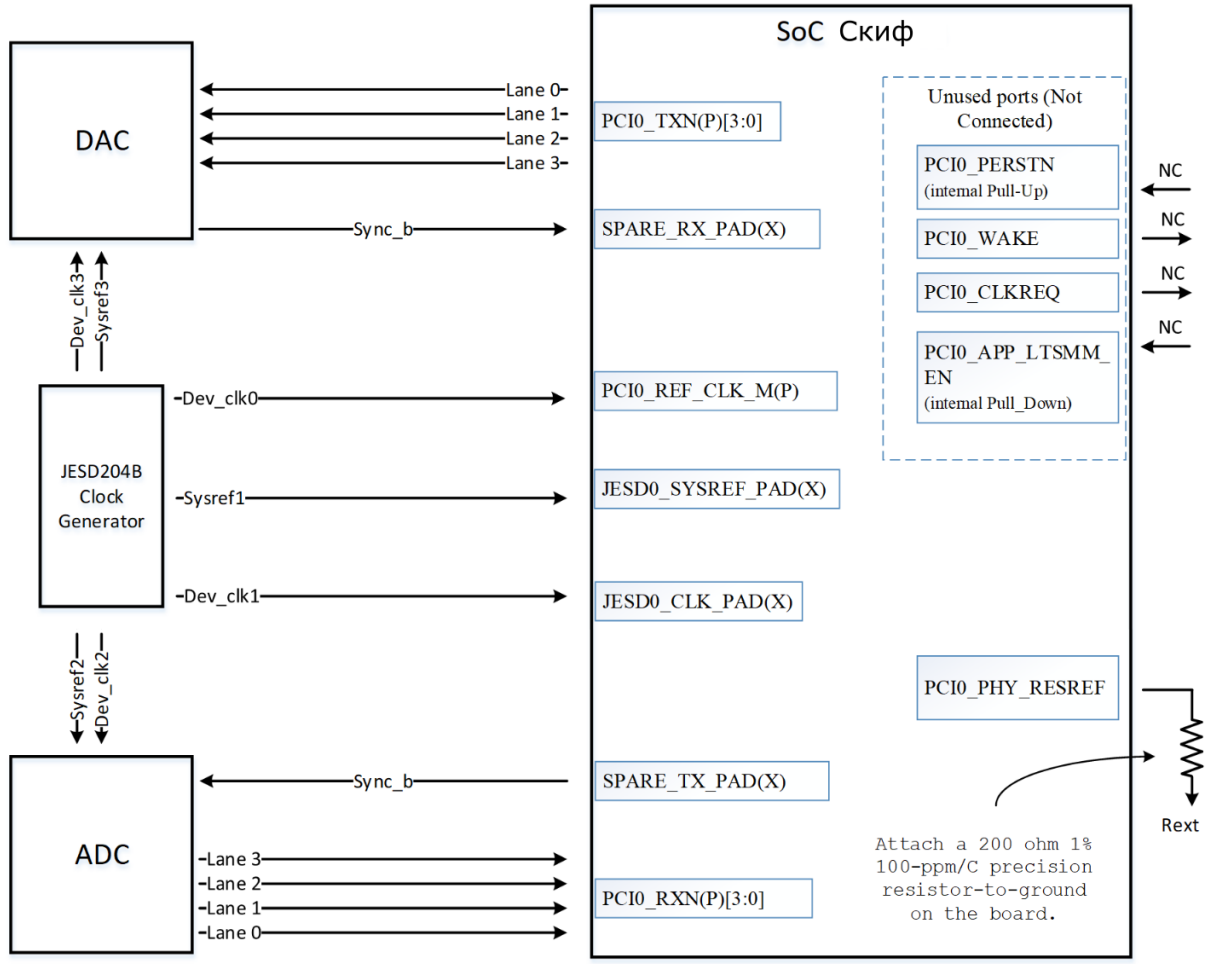


Рисунок 52.2. Подключение к 0-му контроллеру Jesd204b

Аналогично подключение к 1-му контроллеру Jesd204b

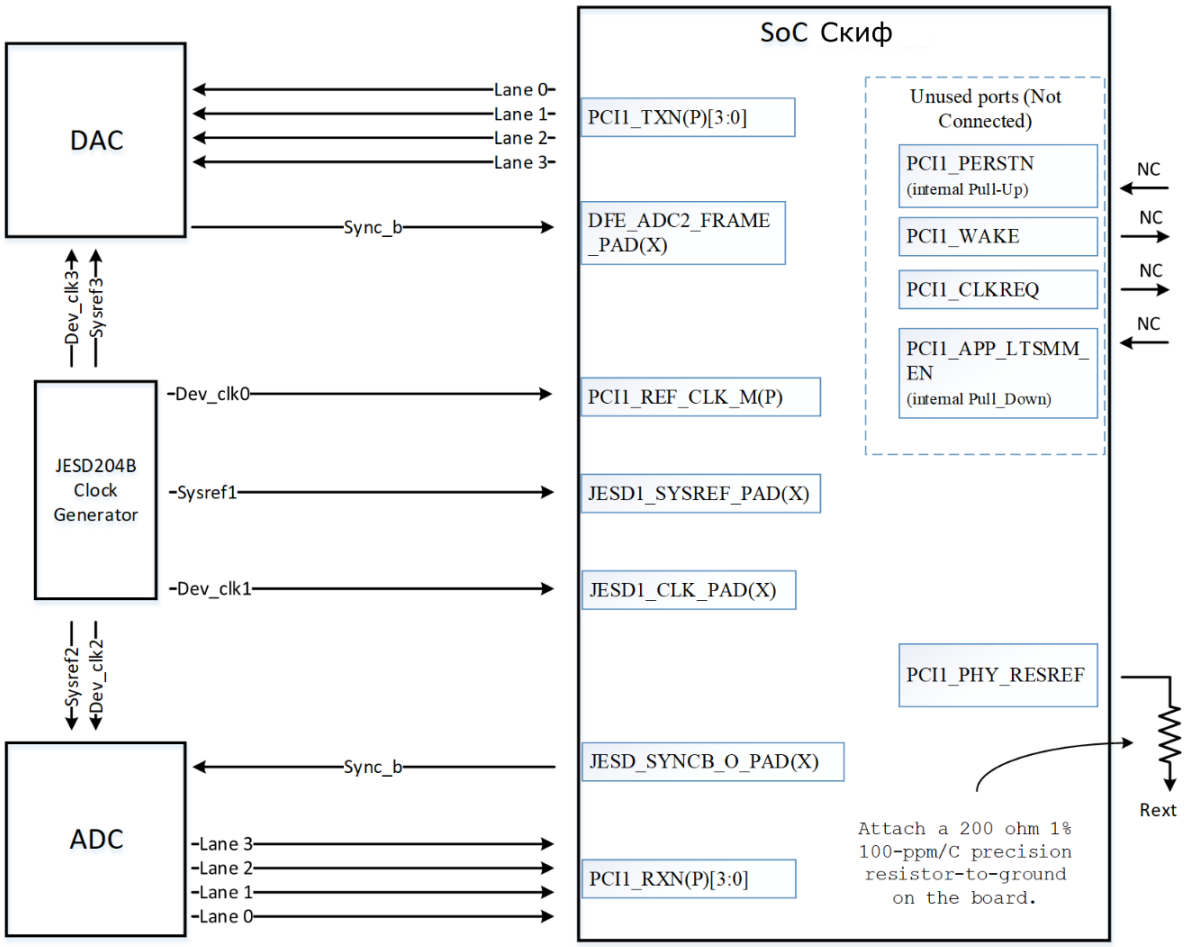


Рисунок 52.3. Подключение к 1-му контроллеру Jesd204b

52.6 NAND. Примеры подключения

Контроллер NAND поддерживает подключение до двух 8-разрядных флэш или до двух 16-разрядных флэш. Поддерживается только последовательная работа с флэш (поле регистра Memory_Address_Register2.mode всегда должно быть установлено в 2'b00). В 8-ми разрядном режиме используются только разряды NFC_IO(7:0), разряды NFC_IO(15:8) необходимо оставить неподключенными.

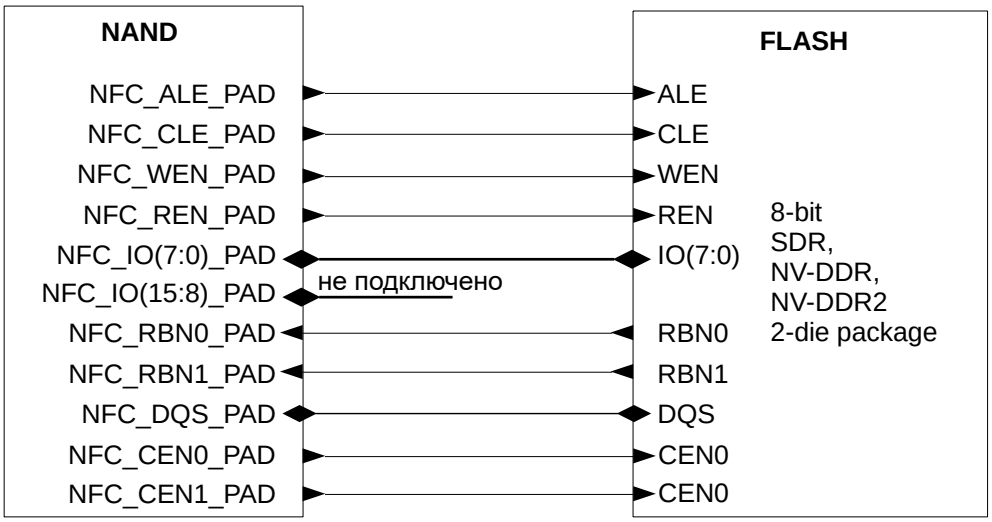


Рисунок 52.4. NAND - подключение флэш (8-bit, 2-die)

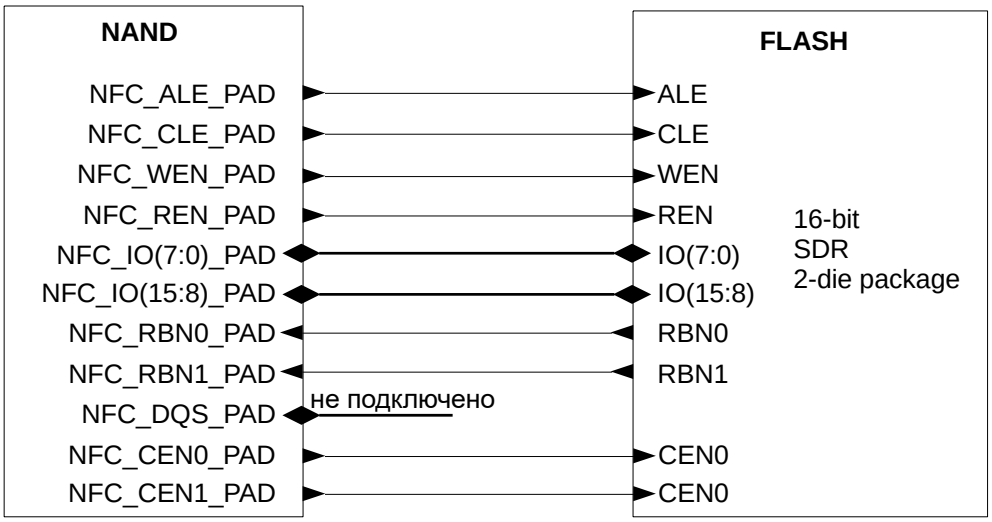


Рисунок 52.5. NAND - подключение флэш (16-bit, 2-die)

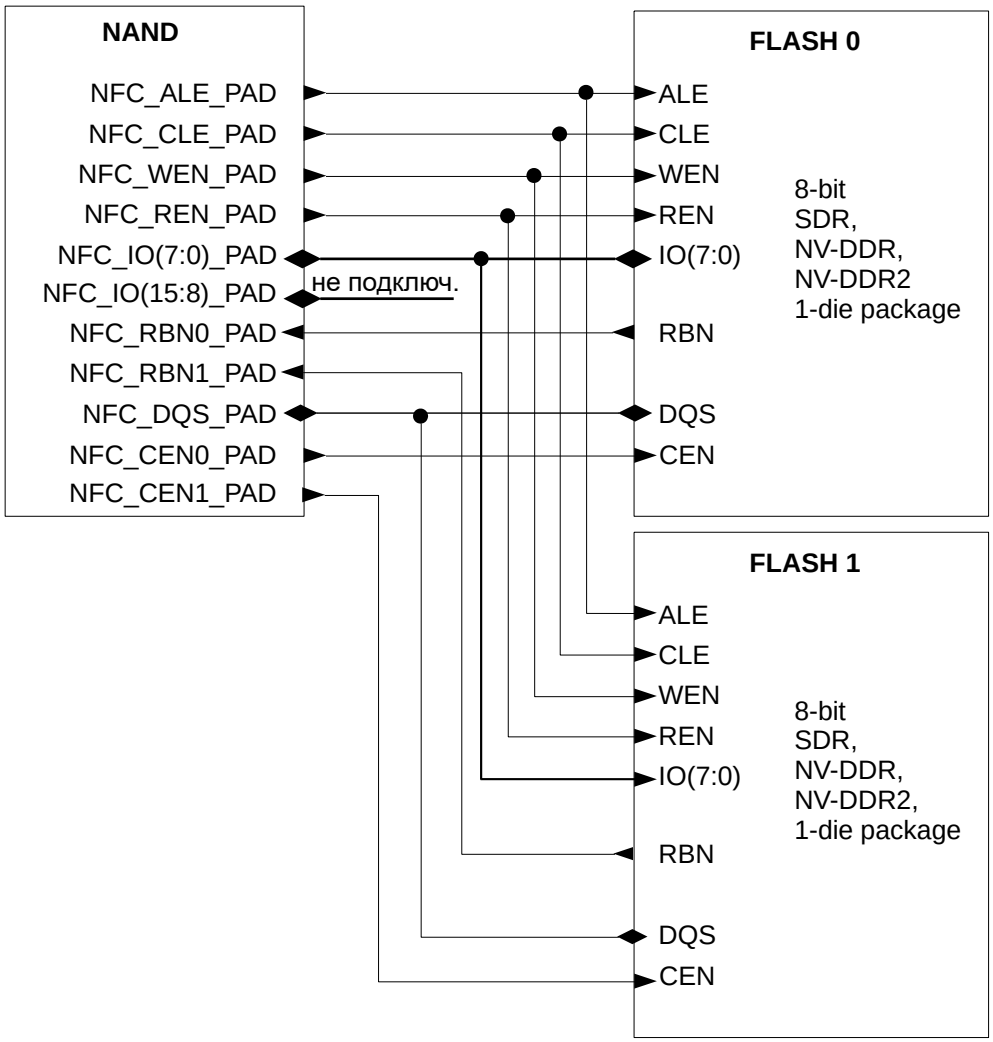


Рисунок 52.6. NAND - подключение 2-х флэш (8-bit)

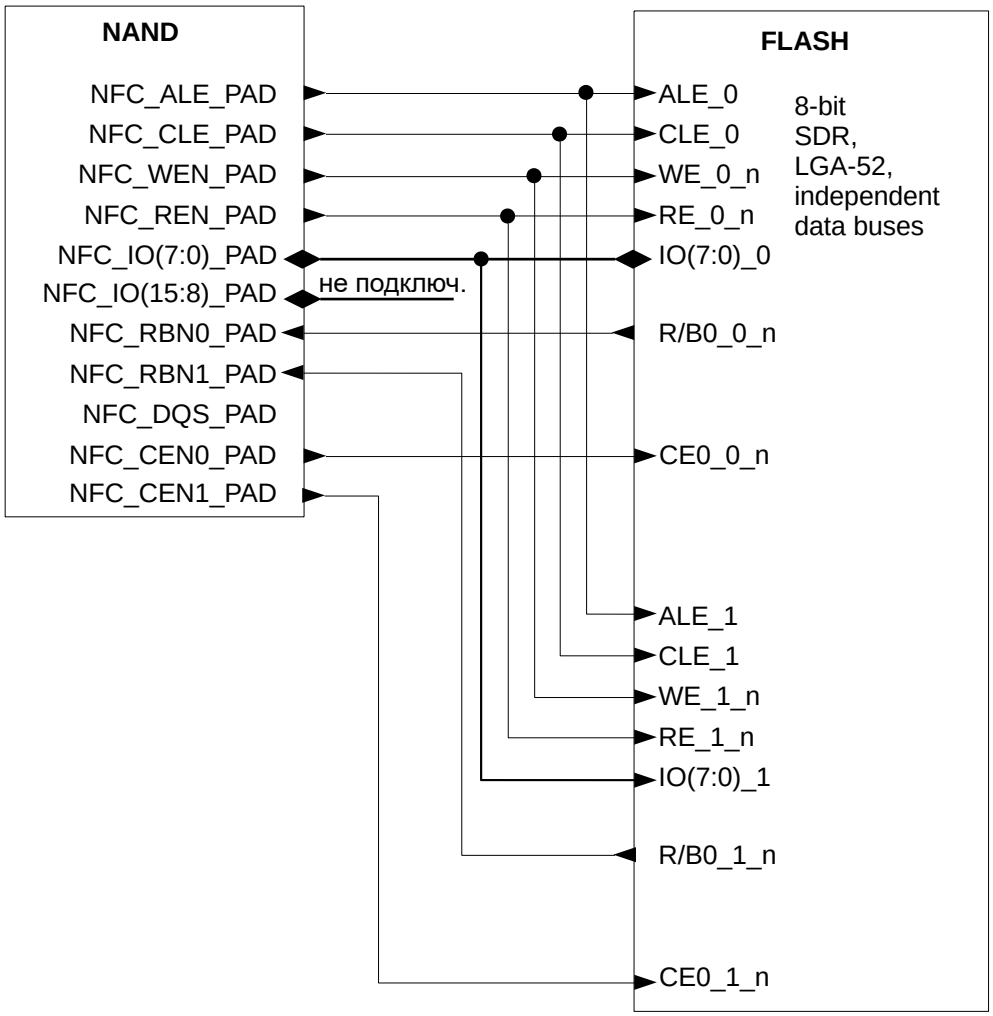


Рисунок 52.7. NAND – подключение флэш (8-bit, 2-die, independent data bus)

52.7 USB. Подключение к источнику питания VBUS +5В

На схеме ниже показан пример подключения контроллера USB к источнику питания VBUS +5V для режима хоста.

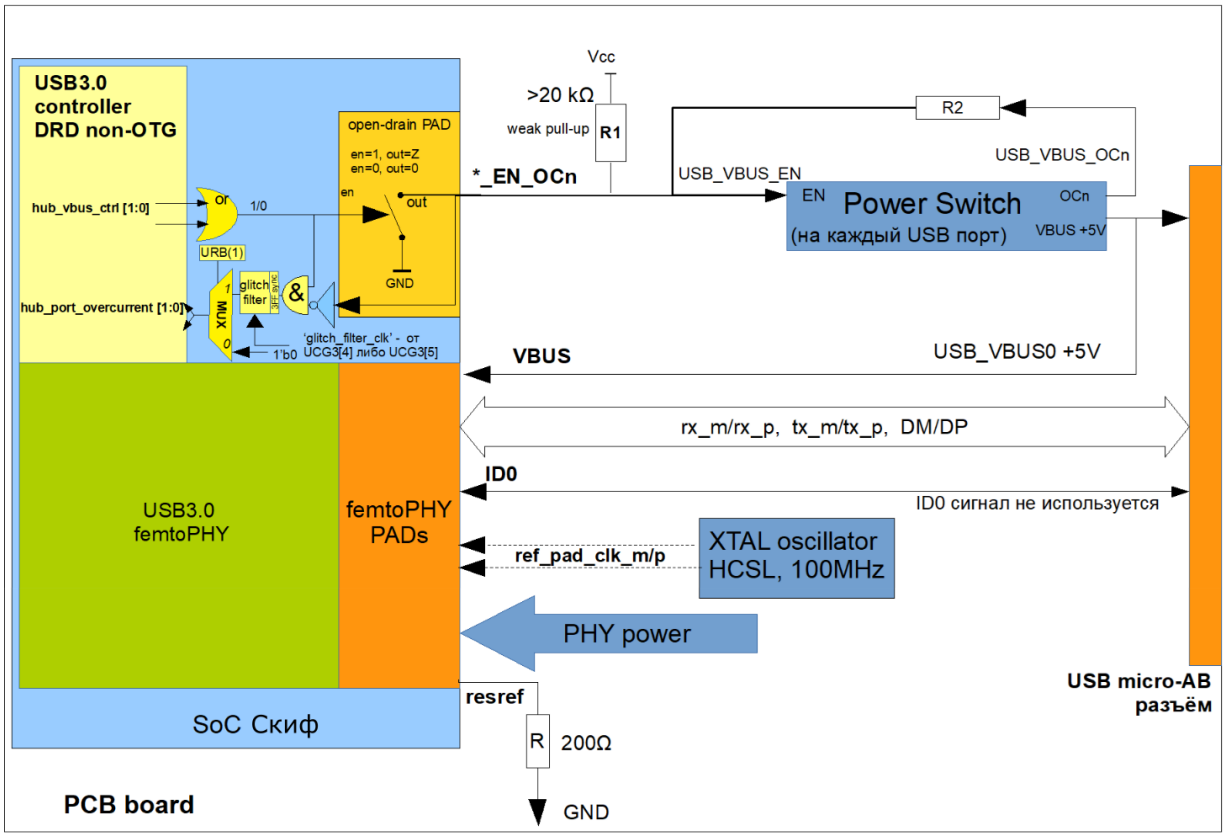


Рисунок 52.8. USB – подключение к источнику VBUS

Выходной сигнал управления включением источника VBUS +5V (см. 'Power Switch' на схеме) – 'hub_vbus_ctrl', а также входной 'hub_port_overcurrent', сигнализирующий о превышении допустимого тока потребления источника VBUS, объединены в один вывод.

Вывод микросхемы USB0_EN_OCN_PAD (USB1_EN_OCN_PAD для второго аналогичного USB3.0) настроен на работу в режиме open-drain.

Предполагается, что выход OCn 'Power Switch' также работает в режиме open-drain.

Если контроллеру USB необходимо включить питание VBUS, то 'hub_vbus_ctrl' устанавливается в '1' и значение USB0_EN_OCn установится в '1' за счет внешней подтяжки ('weak pull-up').

Если контроллеру USB необходимо отключить питание VBUS, то 'hub_vbus_ctrl' устанавливается в '0' и значение USB0_EN_OCn установится в '0'.

При превышении допустимого тока потребления источника VBUS выход OCn 'Power Switch' устанавливается в '0', это приводит к установке в '0' вывода USB0_EN_OCn.

Инвертированное и отфильтрованное внутри микросхемы значение USB0_EN_OCn заведено на вход контроллера hub_port_overcurrent.

При обнаружении превышения по току контроллер автоматически переведёт 'hub_vbus_ctrl' в '0' и будет удерживать его до тех пор, пока ПО не возобновит работу.

Приведенные на схеме номиналы резисторов являются ориентировочными и выбираются разработчиками печатных плат.

52.8 SDMMC. Примеры подключения

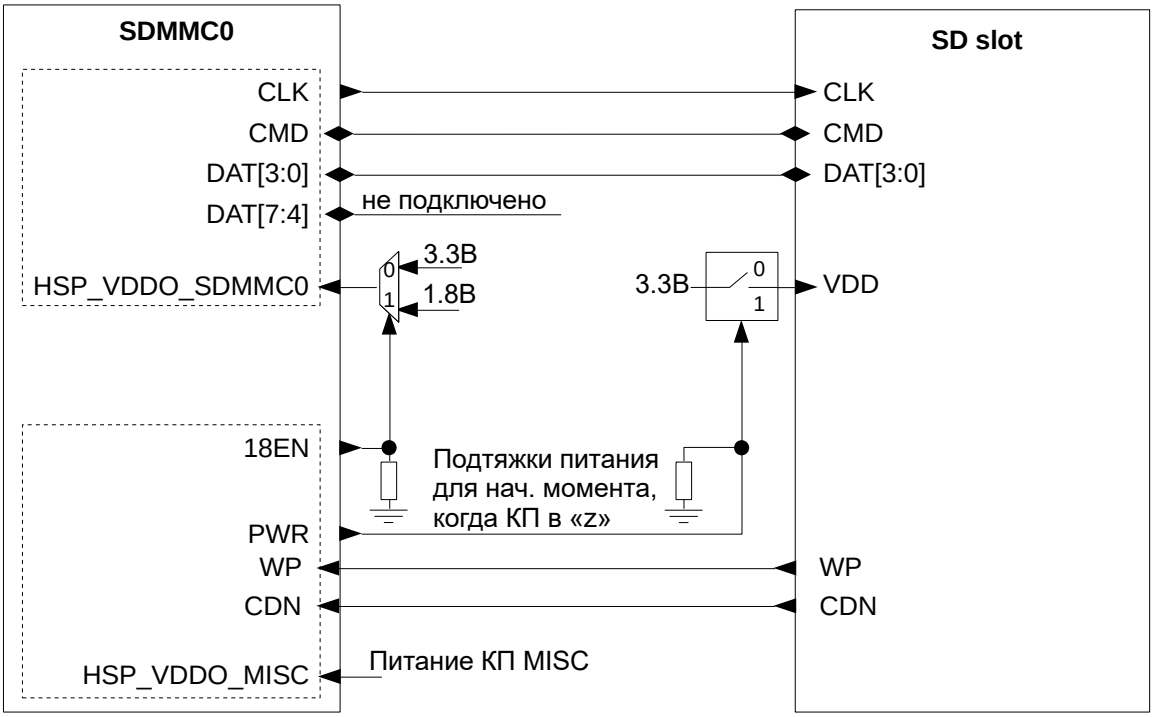


Рисунок 52.9. SDMMC - подключение SD

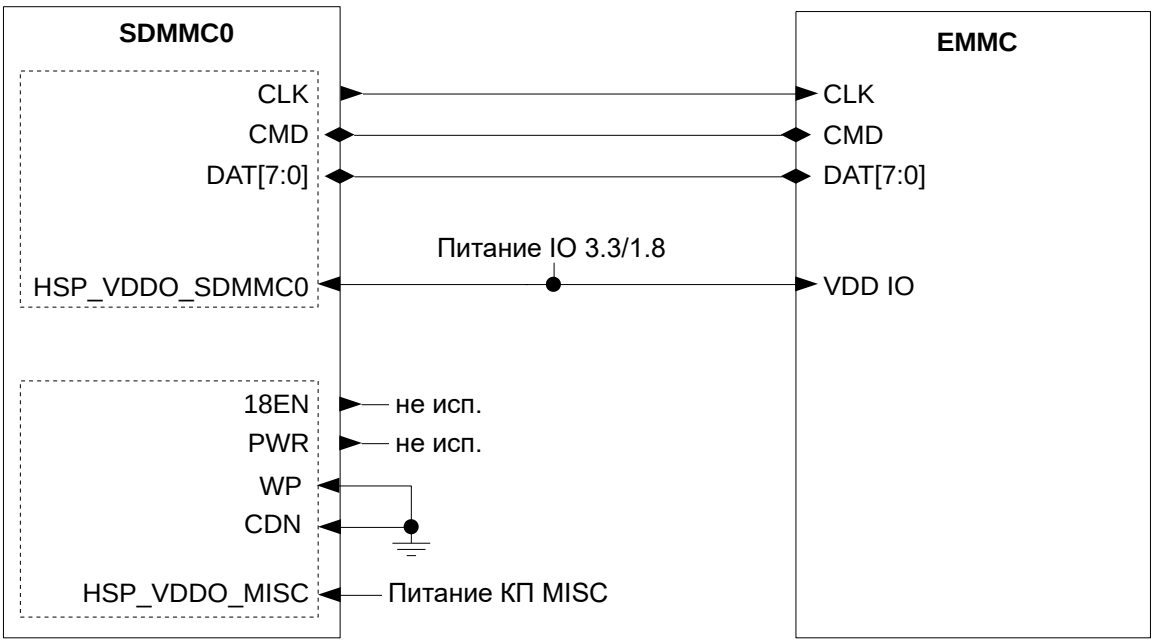


Рисунок 52.10. SDMMC - подключение EMMC

52.9 PCIe и USB. Примеры подключения

Контроллер PCI и USB должны быть подключены к синтезатору частоты, который работает по протоколу HCSL.

Терминация осуществляется с помощью опорного резистора вне микросхемы. Подключение может быть сконфигурировано в заземленном или плавающем режиме, используя rxX_term_acdc (0=плавающая терминация, 1=заземленная терминация), как показано на рисунке ниже.

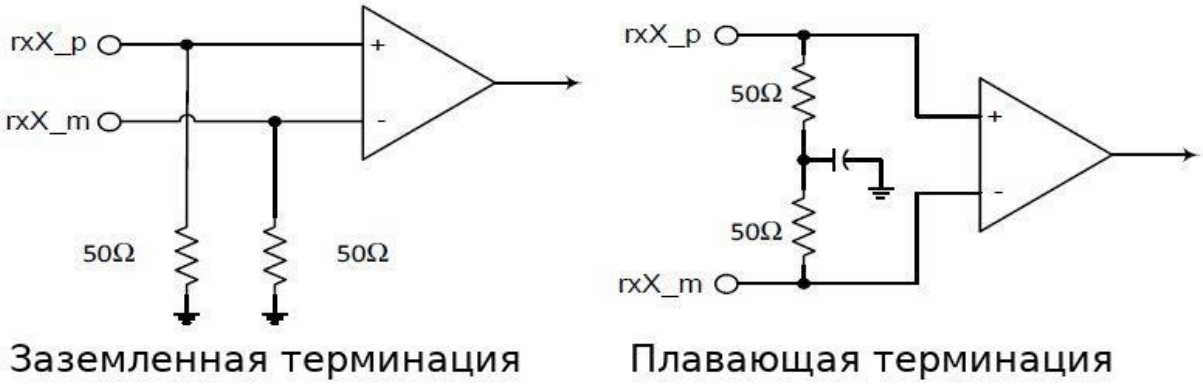
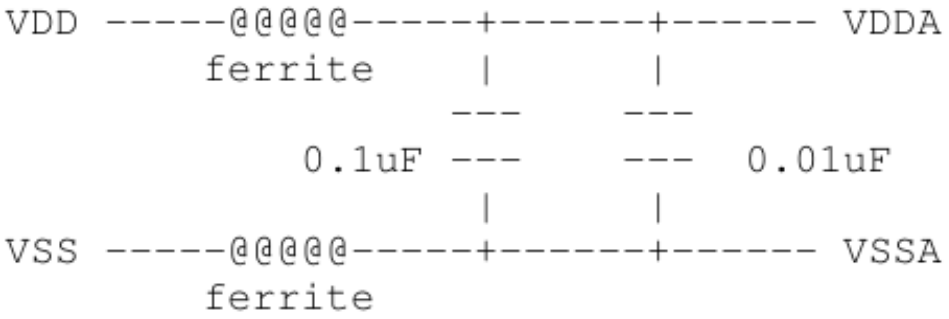


Рисунок 52.11 Пример заземленной и плавающей терминации

52.10 PLL. Рекомендации по подключению земли/питания

Источники аналогового питания PLL должны быть отфильтрованы с помощью двух последовательных ферритовых шариков и двух шунтирующих конденсаторов емкостью 0,1 мкФ и 0,01 мкФ. Феррит на VSS предпочтителен, но необязателен. Добавление феррита в VSS преобразует шум источника питания в шум подложки, видимый PLL. PLL спроектирована так, чтобы быть относительно нечувствительными к шумам источника питания и подложки, поэтому наличие этого феррита является проблемой второго порядка.



Ферритовые бусины должны быть аналогичны одним из следующих от Murata:

Part number		R@DC	Z@10MHz	Z@100MHz	Z@1GHz	size
BLM18EG601SN1	*	0.35	200	600		0603
BLM18PG471SN1		0.20	130	470		0603
BLM18KG601SN1		0.15	160	600		0603
BLM18AG601SN1		0.38	180	600		0603
BLM18AG102SN1		0.50	280	1000		0603
BLM18TG601TN1		0.45	190	600		0603
BLM15AG601SN1		0.60	200	600		0402
BLM15AX601SN1	*	0.34	190	600		0402
BLM15AX102SN1		0.49	250	1000		0402
BLM03AX601SN1		0.85	120	600		0201

* - наиболее предпочтительны

Подобные ферритовые бусины также доступны от Panasonic.

Основные характеристики, которые стоит выбрать:

- сопротивление постоянному току менее 0,40 Ом
- сопротивление на частоте 10 МГц равно или больше 180 Ом
- импеданс на частоте 100 МГц равен или превышает 600 Ом

Конденсаторы следует устанавливать как можно ближе к выводам корпуса.

52.11 PCIe. Пример подключения к слоту

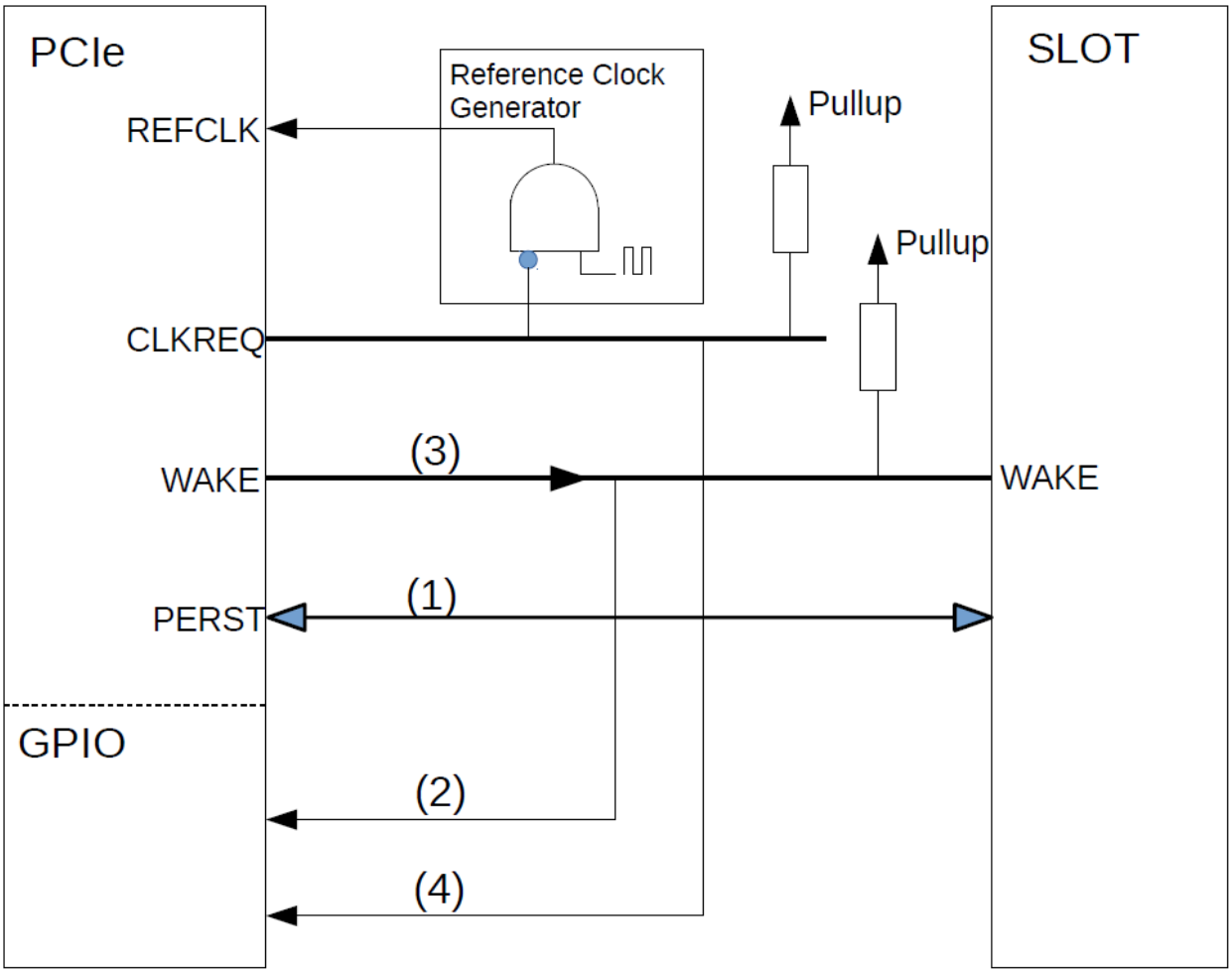


Рисунок 52.12 Пример подключения к слоту

(1) – В режиме RC используется как выход, настраивается через URB регистры. В режиме EP сигнал PERST берется со слота

(2) – В режиме RC идет со слота, в режиме EP не используется

(3) – В режиме RC не используется, PAD переведен в третье состояние. В режиме EP управляет сигналом WAKE слота

(4) – В режиме EP не используется

52.12 Требования к работе OTP

Характеристики OTP на текущем этапе не подтверждены экспериментально, приведенные ниже данные являются теоретическими.

- При программировании OTP необходимо обеспечить подачу напряжения на выводы VPP и VQQ. Если программирование OTP не требуется, выводы могут оставаться неподключенными.

- VPP: минимальное значение напряжения – 5,25 В, максимальное значение – 5,75 В.
- VQQ: напряжение = 2,75 В ± 10%.

53. ИСТОРИЯ ИЗМЕНЕНИЙ

53.1 Версия от 01.07.2025

Обновлены:

- Раздел 2:
  - Обновлена структурная схема микросхемы. В сервисной подсистеме добавлен MAILBOX0, I2C4, удален SPRAM0. В SDR подсистеме добавлены SDRDMA0 и SDRDMA1
  - Список подсистем и блоков микросхемы. В сервисной подсистеме добавлен MAILBOX0, I2C4, удален SPRAM0. В SDR подсистеме добавлены SDRDMA0 и SDRDMA1.

53.2 Версия от 11.07.2025

Добавлено:

- Раздел 24.4.1 “Соответсвие AXI портов мастерам микросхемы”

53.3 Версия от 11.07.2025

53.4 Версия от 29.06.2026

Исправлено:

- Опечатки в таблице 51.22;
- Опечатки в таблицах главы 28;
- Опечатки в заголовке раздела 7.3.21;
- Нумерация таблиц в главе 28;
- Текст в описании регистров COH_AXI_DLOCK и COH_REMAP в главе 3;
- Схема подключения сигналов на рисунке 21.78;
- Назначение 1 вывода GPIO0_PORTC_3;
- Описание формирование пакета MAC Flow-Control (PAUSE) в главе 27;
- Наименования сообщений PCIe и названия регистров в таблице 25.9;

- Тип пакета в главе 27.5.7.1 «Обзор PTP»;
- Грамматические ошибки в таблицах 27.104 и 27.106;
- Неверные типичные UDP Ports в Таблице 27.80;
- Описание сигналов UART3 в таблице 51.14;
- Описание полей регистров fuse0 и fuse1 (fuse0.lock, fuse1.lock);
- Описание бита sys_int в разделе 25.2.1.2 (отправка только INTA).

Удалено:

- Разделы 3.4.1 и 52.10 с упоминанием скрипта TCITSMCN28HPCPCGHPLLA1_calc.csh;
- Цветовое обозначение в таблице 3.15 отображения прерываний.

53.5 Версия от 03.07.2026

Исправлено:

1. Форматирование в некоторых главах;
2. Мелкие опечатки;
3. Формат таблиц в главе 28;
4. Несогласованность единиц измерения.

Удалено:

5. Упоминание V2505 Technical Reference Manual.

53.6 Версия от 10.07.2026

1. Уточнена работа VITDEC;
2. Исправлено описание начального состояния WDT0;
3. Описан максимальный поддерживаемый объем LPDDR4;
4. Уточнены отличия NRST_PON и NRST_WRM;
5. Исправлена разрядность регистров QSPI gpoClr и gpoSet;
6. Унифицирован формат таблиц в главах 18, 28 и 40;

7. Уточнены условия записи в регистры управления сбросом;
8. Уточнены некоторые ссылки на регистры;
9. Исправлен формат ссылок на главы и разделы.

53.7 Версия от 20.07.2026

1. Исправлено форматирование таблиц;
2. Отформатированы отдельные главы.

53.8 Версия от 24.07.2026

Уточнено:

1. Детали синхронизации каналов при настройке частот;
2. Описание VTEST;
3. Ограничения в DFE ALT;

Исправлено:

4. Смещения и размеры полей OTP;
5. Описание USB ID пинов.

53.9 Версия от 31.07.2026

1. Исправлены некорректные адреса карты внутренней памяти подсистемы;
2. Уточнены детали настройки PLL;
3. Добавлено упоминание об отсутствии режима LPI для UCG;
4. Добавлена информация о поддержке 4K@30 при использовании DSI интерфейса;
5. Добавлена информация по требованиям к работе с OTP.

53.10 Версия от 01.07.2026

1. Исправлено форматирование некоторых схем.

53.11 Версия от 31.08.2026

1. Исправлен формат ссылок внутри документа для некоторых глав;
2. Исправлен формат некоторых схем.