

Микросхема интегральная

5023BC016

Техническое описание

Версия 2.04 от 01.09.2017 г.

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Москва, 2017

СОДЕРЖАНИЕ

Введение.....	5
1 Назначение и основные технические характеристики	6
1.1 Архитектурные особенности микросхемы	6
1.2 Конструктивные параметры микросхемы	8
1.3 Электрические параметры микросхемы	19
2 Общая характеристика СБИС CPU	23
3 Описание микросхемы	24
3.1 Структурная организация микросхемы	24
3.1.1 Конфигурационные выводы микросхемы	26
3.1.2 Инициализация работы микросхемы	26
3.2 Распределение адресного пространства СБИС CPU	27
3.3 Вектора прерываний СБИС CPU	29
3.3.1 Регистры INTR_MUX_CTRL	29
3.4 Регистры общего назначения	30
3.4.1 Состав регистров общего назначения	30
3.5 Контроллер внутренней памяти	44
3.6 Контроллер внешней памяти	45
3.7 Контроллер портов ввода-вывода	48
3.7.1 Генерация прерываний	49
3.7.2 Доступ по маске	49
3.7.3 Регистры контроллера GPIO	51
3.8 Контроллер последовательного интерфейса в соответствии с ГОСТ 52070	54
3.8.1 Общие положения	54
3.8.2 Регистры контроллера	55
3.8.3 Регистры контроллера шины	57
3.8.4 Регистры оконечного устройства	60

Инв. № подл.	Изм.	Лист	№ докум.	Подп.	Дата	<div>Версия 2.04 от 01.09.2017 г.</div>		
Инв. № подл.	Изм.	Лист	№ докум.	Подп.	Дата	<div>Микросхема интегральная</div> <div>5023BC016.</div> <div>Техническое описание</div>		
Инв. № подл.	Изм.	Лист	№ докум.	Подп.	Дата	<div>Лит.</div> <div>Лист</div> <div>Листов</div>		
Инв. № подл.	Изм.	Лист	№ докум.	Подп.	Дата	<div>Лит.</div> <div>Лист</div> <div>Листов</div>		
Инв. № подл.	Изм.	Лист	№ докум.	Подп.	Дата	<div>Лит.</div> <div>Лист</div> <div>Листов</div>		

3.8.5 Общие регистры настроек	61
3.8.6 Регистры монитора	63
3.8.7 Регистр INTERRUPT	64
3.8.8 Описание функционирования контроллера последовательного интерфейса по ГОСТ Р 52070	64
3.9 Контроллер интерфейса SpaceWire	72
3.9.1 Описание функционирования контроллера интерфейса SpaceWire	72
3.9.2 Контроллер BUS MASTER	75
3.9.3 Регистры контроллера интерфейса SpaceWire	76
3.10 Контроллер прямого доступа к памяти	80
3.10.1 Общее описание	80
3.10.2 Описание функционирования контроллера DMA	81
3.10.3 Программное управление модулем	98
3.11 Контроллер интерфейса SPI	127
3.11.1 Общие положения	127
3.11.2 Основные характеристики интерфейса SPI	128
3.11.3 Программируемые параметры интерфейса SPI	128
3.11.4 Характеристики интерфейса SPI фирмы Motorola	129
3.11.5 Функциональное описание интерфейса SPI	130
3.11.6 Функционирование модуля SPI	132
3.11.7 Интерфейс прямого доступа к памяти	147
3.11.8 Программное управление модулем	149
3.11.9 Прерывания модуля	156
3.12 Контроллер интерфейса UART	158
3.12.1 Общие положения	158
3.12.2 Основные характеристики интерфейса UART	158
3.12.3 Описание функционирования интерфейса UART	162
3.12.4 Описание функционирования модуля UART	165
3.12.5 Прерывания.....	174
3.12.6 Программное управление модулем	177
3.13 Таймер общего назначения	192

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>					Лист
Изм.	Лист	№ докум.	Подп.	Дата						3

3.13.1	Общее описание	192
3.14	Сторожевой таймер	193
3.14.1	Описание сторожевого таймера	193
3.14.2	Регистры сторожевого таймера (Watchdog)	193
3.15	Передачик телеметрической информации	197
3.15.1	Описание структуры передатчика телеметрической информации	197
3.15.2	Описание работы интерфейса передатчика телеметрической информации	198
3.15.3	Программная модель управления передатчиком телеметрической информации	205
3.15.4	Регистры передатчика телеметрической информации	207
3.16	Приёмник телекомандной информации	211
3.16.1	Общее описание приёмника телекомандной информации	211
3.16.2	Описание работы интерфейса приёмника телекомандной информации	212
3.16.3	Программная модель управления приёмником телекомандной информации	216
3.16.4	Описание регистров приёмника телекомандной информации	219
3.17	Специализированный вычислитель с плавающей точкой	222
3.17.1	Описание регистров	223
3.18	Контроллер интерфейса CAN	225
3.18.1	Общее описание	225
3.18.2	Базовый режим работы	226
3.18.3	Расширенный режим работы	231
3.18.4	Общие регистры для базового и расширенного режимов работы	243
3.19	Контроллер интерфейса I ² C	245
3.19.1	Описание работы блока I ² C-мастер	246
3.19.2	Описание регистров блока I ² C-мастер	252
	Перечень сокращений и условных обозначений	255
	Перечень ссылочных документов	257
	Перечень изменений	258

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>					Лист
Изм.	Лист	№ докум.	Подп.	Дата						4

Введение

Настоящее техническое описание распространяется на интегральную микросхему 5023BC016 (далее — микросхема СБИС CPU), предназначенную для применения в качестве специализированной схемы обработки информации в радиоэлектронной аппаратуре специального назначения.

Радиационно-стойкая специализированная микросхема 5023BC016 предназначена для построения аппаратуры контрольно-измерительных систем (КИС), телеметрии служебных систем космических аппаратов с учетом эксплуатации их в жестких условиях специальных видов воздействующих факторов.

Микросхема может использоваться для организации каналов связи по интерфейсам магистральной последовательной шины по ГОСТ Р 52070 с резервированием.

При этом микросхема может использоваться в качестве контроллера шины (КШ), оконечного устройства (ОУ), а также монитора.

Наличие двух контроллеров интерфейсов SpaceWire со скоростью передачи данных до 100 Мбит/с позволяет использовать микросхему в каналах передачи данных данного типа.

Микросхема имеет блоки передатчика телеметрической информации и приёмника телекомандной информации, позволяющие использовать различные виды кодирования (LDPC, Турбо, Рида-Соломона).

Применение микросхемы позволяет создавать системы управления космическими аппаратами совместимыми с международными стандартами CCSDS, уменьшить массогабаритные характеристики и потребляемую мощность аппаратуры КИС, телеметрии служебных систем космических аппаратов в 4 – 5 раз и обеспечить требования Федерального космического агентства по дозовой стойкости и воздействию тяжелых заряженных частиц.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	<div>Микросхема интегральная 5023BC016</div> <div>Техническое описание</div> <div>Версия 2.04 от 01.09.2017 г.</div>					Лист
Изм	Лист	№ докум.	Подп.	Дата						5

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.

Интегральная микросхема 5023BC016 представляет собой однокристалльный 32 разрядный процессор, выполненный по КМОП технологии с нормами 0,18 мкм. Микросхема предназначена для построения аппаратуры КИС, телеметрии служебных систем космических аппаратов с учетом эксплуатации их в жестких условиях специальных видов воздействующих факторов.

разрядность шины адреса ядра процессора.....	32 бита
конвейер	3 стадии
аппаратный умножитель	32 x 32
тип встроенной шины	AMBA
размер встроенного оперативного запоминающего устройства (ОЗУ).....	128 кБ
тактовая частота, МГц	50 ¹⁾

напряжение питания ядра, В $1,8 \pm 10 \%$

напряжение питания буферов ввода-вывода, В..... $3,3 \pm 10 \%$

СБИС CPU включает следующие основные блоки:

а) ядро 32-битного Reduced Instruction Set Computer (RISC) процессора;

б) четыре независимых контроллера последовательного интерфейса по ГОСТ Р 52070 с резервированием;

в) два независимых контроллера интерфейса SpaceWire со скоростью передачи данных до 100 Мбит/с;

г) контроллер прямого доступа к памяти;

д) внутрикристальную схему отладки с интерфейсом Joint Test Action Group (JTAG) (IEEE 1149.1);

¹⁾ Возможна работа на тактовой частоте 80 МГц. При этом значения динамического тока потребления ядра для основных режимов работы микросхемы приведены в таблице 1.3.3.

е) ОЗУ размером 128 кБ со встроенным блоком обнаружения и исправления ошибок;

ж) контроллер внешней памяти со встроенным блоком обнаружения и исправления ошибок, с поддержкой до четырех микросхем внешней памяти и с поддержкой кэширования;

з) шесть независимых контроллеров асинхронного последовательного интерфейса Universal asynchronous receiver-transmitter (UART);

и) два независимых контроллера последовательного интерфейса Serial Peripheral Interface (SPI);

к) контроллер приёмника телекомандной информации (ПрТКИ) по стандарту Consultative Committee for Space Data Systems (CCSDS), включающий следующие блоки:

1) декодер БЧХ (63,56) — размер блока 63 бита, размер полезных данных 56 бит;

2) декодер Витерби;

3) дерандомизатор;

л) контроллер передатчика телеметрической информации (ПдТМИ) по стандарту CCSDS, включающий следующие блоки:

1) энкодер Рида-Соломона с параметрами ($N = 255$, $K = 223$), ($N = 255$, $K = 239$) — размер блока 255 символов, размер полезных данных 223 или 239 символов, разрядность 8 бит, порождающий полином $F(x) = x^8 + x^7 + x^2 + x + 1$;

2) турбо энкодер со скоростью кодирования (отношению объема полезных данных к объёму передаваемых) 1/2, 1/3, 1/4, 1/6;

3) LDPC энкодер с параметрами (8160, 7136) (размер кодового блока 8160 символов, размер блока данных 7136 символов);

4) сверточный энкодер со скоростью кодирования (отношению объема полезных данных к объёму передаваемых) 1/2, 2/3, 3/4, 5/6 или 7/8, разрядность 7 бит, порождающие полиномы: $G1 = 1111001$, $G2 = 1011011$;

5) рандомизатор;

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>					Лист
Изм.	Лист	№ докум.	Подп.	Дата						7

- б) формирователь кадров;
- м) два независимых контроллера интерфейса Controller Area Network (CAN);
- н) контроллер интерфейса I²C в режиме мастера;
- о) специализированный вычислитель с плавающей точкой.

1.2 Конструктивные параметры микросхемы

Функциональное назначение выводов микросхемы приведено в таблице 1.2.1.

Микросхема выполнена в металлокерамическом корпусе 6112.145-1 и предназначена для ручной и автоматической сборки в соответствии с ГОСТ В 20.39.412.

Масса микросхемы не более 20 г.

Значение собственной резонансной частоты не менее 3,6 кГц.

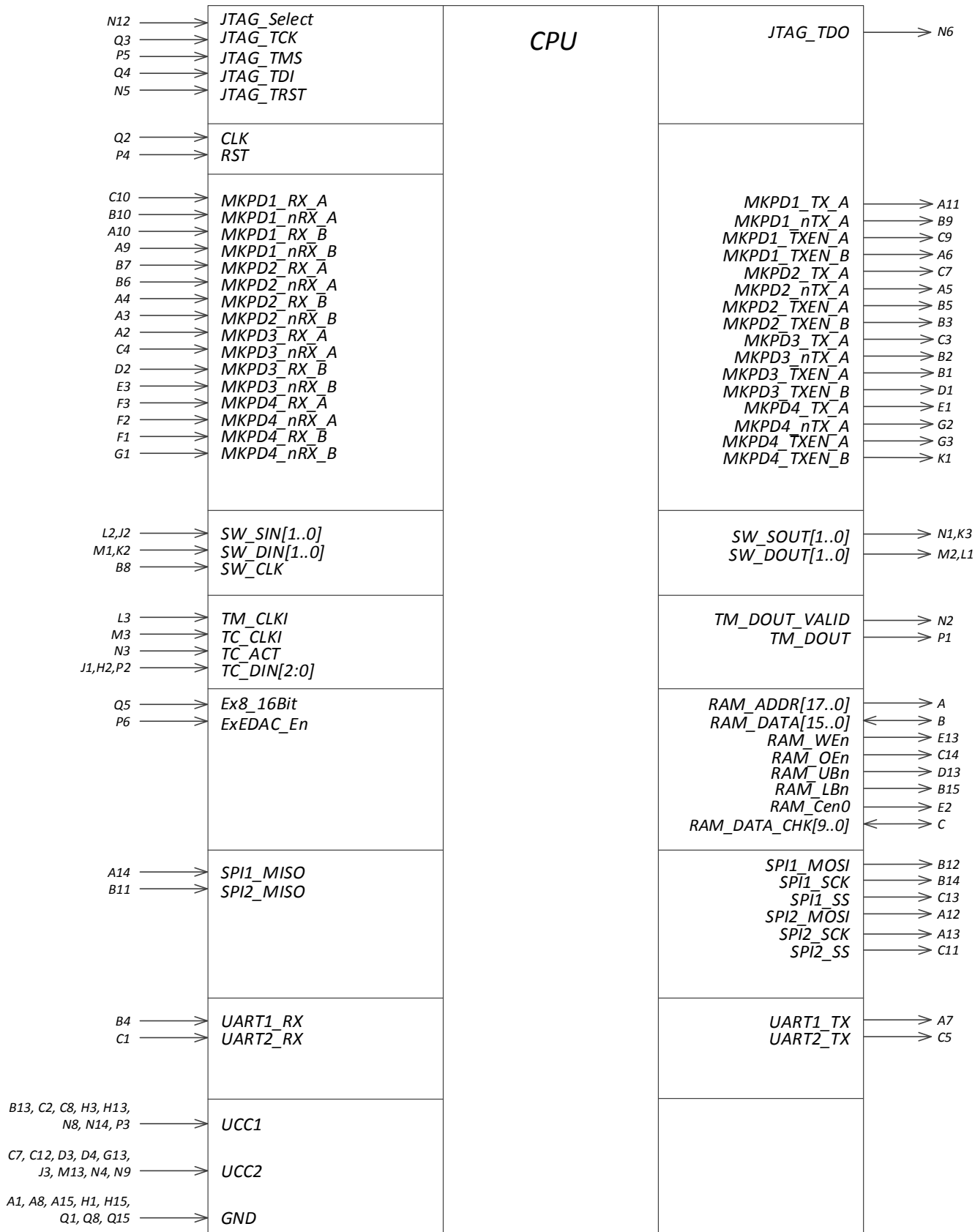
Предельная повышенная температура среды плюс 125 °С.

Показатель герметичности микросхем со свободным внутренним объёмом по эквивалентному нормализованному потоку не более $6,65 \cdot 10^{-3} \text{ Па} \cdot \text{см}^3/\text{с}$.

Более подробно конструктивные характеристики микросхемы приведены в технических условиях АЕЯР.431290.994 ТУ.

Условное графическое обозначение микросхемы приведено на рисунке 1.2.1.

Ине. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>					Лист
Изм	Лист	№ докум.	Подп.	Дата						8



ПримечаниеА: Q14,P13,N11,P12,Q13,Q12,P11,Q11,N10,P10,P9,Q10,Q9,P8,Q7,Q6,N7,P7
 В: H14,J15,K15,J13,J14,L15,K14,K13,M15,L14,N15,L13,M14,P15,P14,N13
 С: D14,C15,D15,E14,E15,F13,F14,G14,F15,G15

Рисунок 1.2.1 — Условное графическое обозначение СБИС CPU

Таблица 1.2.1 — Функциональное назначение выводов

Номер вывода	Название вывода	Тип вывода	Номер соответствующего порта ввода-вывода/функция
1	2	3	4
A1	GND	Общий	Питание
A2	MKPD3_RX_A	Вход/выход	GPIO_F(4)
A3	MKPD2_nRX_B	Вход/выход	GPIO_F(2)
A4	MKPD2_RX_B	Вход/выход	GPIO_F(1)
A5	MKPD2_nTX_A	Вход/выход	GPIO_E(15)
A6	MKPD1_TXEN_B	Вход/выход	GPIO_E(11)
A7	UART1_TxD	Вход/выход	GPIO_D(15)
A8	GND	Общий	Питание
A9	MKPD1_nRX_B	Вход/выход	GPIO_E(10)
A10	MKPD1_RX_B	Вход/выход	GPIO_E(9)
A11	MKPD1_TX_A	Вход/выход	GPIO_E(6)
A12	SPI2_MOSI	Вход/выход	GPIO_D(14)
A13	SPI2_SCK	Вход/выход	GPIO_D(12)
A14	SPI1_MISO	Вход/выход	GPIO_D(9)
A15	GND	Общий	Питание
B1	MKPD3_TXEN_A	Вход/выход	GPIO_F(8)
B2	MKPD3_nTX_A	Вход/выход	GPIO_F(7)
B3	MKPD2_TXEN_B	Вход/выход	GPIO_F(3)
B4	UART1_RxD	Вход/выход	GPIO_E(0)
B5	MKPD2_TXEN_A	Вход/выход	GPIO_F(0)
B6	MKPD2_nRX_A	Вход/выход	GPIO_E(13)
B7	MKPD2_RX_A	Вход/выход	GPIO_E(12)
B8	SW_CLK	Вход/выход	GPIO_H(4)
B9	MKPD1_nTX_A	Вход/выход	GPIO_E(7)
B10	MKPD1_nRX_A	Вход/выход	GPIO_E(5)
B11	SPI2_MISO	Вход/выход	GPIO_D(13)
B12	SPI1_MOSI	Вход/выход	GPIO_D(10)
B13	VCC1	Питание UCC1	Питание

Ине. № подл.	Подп. и дата.
Взам. инв. №	Ине. № дубл.
Подп. и дата	

Изм.	Лист	№ докум.	Подп.	Дата
------	------	----------	-------	------

Микросхема интегральная 5023BC016

Техническое описание
Версия 2.04 от 01.09.2017 г.

Лист

10

Копировал

Формат А4

Продолжение таблицы 1.2.1

1	2	3	4
B14	SPI1_SCK	Вход/выход	GPIO_D(8)
B15	RAM_Lbn	Вход/выход	GPIO_D(5)
C1	UART2_RX	Вход/выход	GPIO_E(1)
C2	VCC1	Питание UCC1	Питание
C3	MKPD3_TX_A	Вход/выход	GPIO_F(6)
C4	MKPD3_nRX_A	Вход/выход	GPIO_F(5)
C5	UART2_TX	Вход/выход	GPIO_E(2)
C6	MKPD2_TX_A	Вход/выход	GPIO_E(14)
C7	VCC2	Питание UCC2	Питание
C8	VCC1	Питание UCC1	Питание
C9	MKPD1_TXEN_A	Вход/выход	GPIO_E(8)
C10	MKPD1_RX_A	Вход/выход	GPIO_E(4)
C11	SPI2_SS	Вход/выход	GPIO_D(11)
C12	VCC2	Питание UCC2	Питание
C13	SPI1_SS	Вход/выход	GPIO_D(7)
C14	RAM_Oen	Вход/выход	GPIO_D(4)
C15	RAM_DATA_CHK(8)	Вход/выход	GPIO_D(0)
D1	MKPD3_TXEN_B	Вход/выход	GPIO_F(11)
D2	MKPD3_RX_B	Вход/выход	GPIO_F(9)
D3	VCC2	Питание UCC2	Питание
D4	VCC2	Питание UCC2	Питание
D13	RAM_Ubn	Вход/выход	GPIO_D(6)
D14	RAM_DATA_CHK(9)	Вход/выход	GPIO_D(1)
D15	RAM_DATA_CHK(7)	Вход/выход	GPIO_C(15)
E1	MKPD4_TX_A	Вход/выход	GPIO_F(14)
E2	RAM_Cen0	Вход/выход	GPIO_E(3)
E3	MKPD3_nRX_B	Вход/выход	GPIO_F(10)
E13	RAM_Wen	Вход/выход	GPIO_D(2)
E14	RAM_DATA_CHK(6)	Вход/выход	GPIO_C(14)

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.

Изм	Лист	№ докум.	Подп.	Дата

Микросхема интегральная 5023BC016
Техническое описание
Версия 2.04 от 01.09.2017 г.

Лист
11

Продолжение таблицы 1.2.1

1	2	3	4
E15	RAM_DATA_CHK(5)	Вход/выход	GPIO_C(13)
F1	MKPD4_RX_B	Вход/выход	GPIO_G(1)
F2	MKPD4_nRX_A	Вход/выход	GPIO_F(13)
F3	MKPD4_RX_A	Вход/выход	GPIO_F(12)
F13	RAM_DATA_CHK(4)	Вход/выход	GPIO_C(12)
F14	RAM_DATA_CHK(3)	Вход/выход	GPIO_C(11)
F15	RAM_DATA_CHK(1)	Вход/выход	GPIO_C(9)
G1	MKPD4_nRX_B	Вход/выход	GPIO_G(2)
G2	MKPD4_nTX_A	Вход/выход	GPIO_F(15)
G3	MKPD4_TXEN_A	Вход/выход	GPIO_G(0)
G13	VCC2	Питание UCC2	Питание
G14	RAM_DATA_CHK(2)	Вход/выход	GPIO_C(10)
G15	RAM_DATA_CHK(0)	Вход/выход	GPIO_C(8)
H1	GND	Общий	Питание
H2	TC_DIN(1)	Вход/выход	GPIO_H(2)
H3	VCC1	Питание UCC1	Питание
H13	VCC1	Питание UCC1	Питание
H14	RAM_DATA(15)	Вход/выход	GPIO_C(7)
H15	GND	Общий	Питание
J1	TC_DIN(2)	Вход/выход	GPIO_H(3)
J2	SW_SIN(0)	Вход/выход	GPIO_G(4)
J3	VCC2	Питание UCC2	Питание
J13	RAM_DATA(12)	Вход/выход	GPIO_C(4)
J14	RAM_DATA(11)	Вход/выход	GPIO_C(3)
J15	RAM_DATA(14)	Вход/выход	GPIO_C(6)
K1	MKPD4_TXEN_B	Вход/выход	GPIO_G(3)
K2	SW_DIN(0)	Вход/выход	GPIO_G(5)
K3	SW_SOUT(0)	Вход/выход	GPIO_G(6)
K13	RAM_DATA(8)	Вход/выход	GPIO_C(0)

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.

Изм	Лист	№ докум.	Подп.	Дата

Микросхема интегральная 5023BC016
Техническое описание
Версия 2.04 от 01.09.2017 г.

Лист
12

Продолжение таблицы 1.2.1

1	2	3	4
K14	RAM_DATA(9)	Вход/выход	GPIO_C(1)
K15	RAM_DATA(13)	Вход/выход	GPIO_C(5)
L1	SW_DOUT(0)	Вход/выход	GPIO_G(7)
L2	SW_SIN(1)	Вход/выход	GPIO_G(8)
L3	TM_CLKI	Вход/выход	GPIO_G(12)
L13	RAM_DATA(4)	Вход/выход	GPIO_B(12)
L14	RAM_DATA(6)	Вход/выход	GPIO_B(14)
L15	RAM_DATA(10)	Вход/выход	GPIO_C(2)
M1	SW_DIN(1)	Вход/выход	GPIO_G(9)
M2	SW_DOUT(1)	Вход/выход	GPIO_G(11)
M3	TC_CLKI	Вход/выход	GPIO_G(15)
M13	VCC2	Питание UCC2	Питание
M14	RAM_DATA(3)	Вход/выход	GPIO_B(11)
M15	RAM_DATA(7)	Вход/выход	GPIO_B(15)
N1	SW_SOUT(1)	Вход/выход	GPIO_G(10)
N2	TM_DOUT_VALID	Вход/выход	GPIO_G(13)
N3	TC_ACT	Вход/выход	GPIO_H(0)
N4	VCC2	Питание UCC2	Питание
N5	JTAG_TRST	Вход	Сброс контроллера интерфейса JTAG
N6	JTAG_TDO	Выход	Выход данных интерфейса JTAG
N7	RAM_ADDR(1)	Вход/выход	GPIO_A(1)
N8	VCC1	Питание UCC1	Питание
N9	VCC2	Питание UCC2	Питание
N10	RAM_ADDR(9)	Вход/выход	GPIO_A(9)
N11	RAM_ADDR(15)	Вход/выход	GPIO_A(15)
N12	JTAG_Select	Вход	Выбор контроллера интерфейса JTAG
N13	RAM_DATA(0)	Вход/выход	GPIO_B(8)
N14	VCC1	Питание UCC1	Питание

Ине. № подл.	Подп. и дата	Взам. инв. №	Ине. № дубл.	Подп. и дата.

Изм	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 13
-----	------	----------	-------	------	---	------------

Продолжение таблицы 1.2.1

1	2	3	4
N15	RAM_DATA(5)	Вход/выход	GPIO_B(13)
P1	TM_DOUT	Вход/выход	GPIO_G(14)
P2	TC_DIN(0)	Вход/выход	GPIO_H(1)
P3	VCC1	Питание UCC1	Питание
P4	RST	Вход	Сброс микросхемы. Активный уровень – «0»
P5	JTAG_TMS	Вход	Выбор режима интерфейса JTAG
P6	ExEDAC_En	Вход	Включение блока исправления ошибок внешней памяти
P7	RAM_ADDR(0)	Вход/выход	GPIO_A(0)
P8	RAM_ADDR(4)	Вход/выход	GPIO_A(4)
P9	RAM_ADDR(7)	Вход/выход	GPIO_A(7)
P10	RAM_ADDR(8)	Вход/выход	GPIO_A(8)
P11	RAM_ADDR(11)	Вход/выход	GPIO_A(11)
P12	RAM_ADDR(14)	Вход/выход	GPIO_A(14)
P13	RAM_ADDR(16)	Вход/выход	GPIO_B(0)
P14	RAM_DATA(1)	Вход/выход	GPIO_B(9)
P15	RAM_DATA(2)	Вход/выход	GPIO_B(10)
Q1	GND	Общий	Питание
Q2	CLK	Вход	Тактовый сигнал цифровой части микросхемы
Q3	JTAG_TCK	Вход	Тактовый сигнал интерфейса JTAG
Q4	JTAG_TDI	Вход	Вход данных интерфейса JTAG
Q5	Ex8_16bit	Вход	Выбор режима работы интерфейса внешней памяти
Q6	RAM_ADDR(2)	Вход/выход	GPIO_A(2)
Q7	RAM_ADDR(3)	Вход/выход	GPIO_A(3)
Q8	GND	Общий	Питание
Q9	RAM_ADDR(5)	Вход/выход	GPIO_A(5)
Q10	RAM_ADDR(6)	Вход/выход	GPIO_A(6)
Q11	RAM_ADDR(10)	Вход/выход	GPIO_A(10)
Q12	RAM_ADDR(12)	Вход/выход	GPIO_A(12)

Ине. № подл.	Подп. и дата	Взам. инв. №	Ине. № дубл.	Подп. и дата.

Изм	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 14
-----	------	----------	-------	------	---	------------

Продолжение таблицы 1.2.1

1	2	3	4
Q13	RAM_ADDR(13)	Вход/выход	GPIO_A(13)
Q14	RAM_ADDR(17)	Вход/выход	GPIO_B(1)
Q15	GND	Общий	Питание

Выходы могут использоваться как в качестве интерфейсных сигналов, так и в режиме General Purpose input-output (GPIO). В режиме GPIO выходы разбиты на порты по 16 выводов в каждом. Управление каждым портом осуществляется с помощью контроллера портов ввода-вывода (п. 3.7).

Каждый вывод может поддерживать от одного до трех различных интерфейсных сигналов в режиме включенной альтернативной функции. Выбор используемой альтернативной функции осуществляется путем записи в регистры ALT_FUNCTION_CTRL в блоке регистров общего назначения. В таблице 1.2.2 приведено соответствие интерфейсных сигналов для различных альтернативных функций, и портов ввода-вывода, описанных в таблице 1.2.1.

Таблица 1.2.2 — Соответствие интерфейсных сигналов и портов ввода-вывода

Номер порта ввода-вывода	ALT_FUNC_1	ALT_FUNC_2	ALT_FUNC_3	PAD
1	2	3	4	5
A[0]	RAM_ADDR[0]			P7
A[1]	RAM_ADDR[1]			N7
A[2]	RAM_ADDR[2]			Q6
A[3]	RAM_ADDR[3]			Q7
A[4]	RAM_ADDR[4]			P8
A[5]	RAM_ADDR[5]			Q9
A[6]	RAM_ADDR[6]			Q10
A[7]	RAM_ADDR[7]			P9
A[8]	RAM_ADDR[8]			P10
A[9]	RAM_ADDR[9]			N10
A[10]	RAM_ADDR[10]			Q11
A[11]	RAM_ADDR[11]			P11
A[12]	RAM_ADDR[12]			Q12
A[13]	RAM_ADDR[13]			Q13

Таблица 1.2.2 — Соответствие интерфейсных сигналов и портов ввода-вывода				
Номер порта ввода-вывода	ALT_FUNC_1	ALT_FUNC_2	ALT_FUNC_3	PAD
1	2	3	4	5
A[0]	RAM_ADDR[0]			P7
A[1]	RAM_ADDR[1]			N7
A[2]	RAM_ADDR[2]			Q6
A[3]	RAM_ADDR[3]			Q7
A[4]	RAM_ADDR[4]			P8
A[5]	RAM_ADDR[5]			Q9
A[6]	RAM_ADDR[6]			Q10
A[7]	RAM_ADDR[7]			P9
A[8]	RAM_ADDR[8]			P10
A[9]	RAM_ADDR[9]			N10
A[10]	RAM_ADDR[10]			Q11
A[11]	RAM_ADDR[11]			P11
A[12]	RAM_ADDR[12]			Q12
A[13]	RAM_ADDR[13]			Q13

Изм.	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист
						15

Продолжение таблицы 1.2.2

1	2	3	4	5
A[14]	RAM_ADDR[14]			P12
A[15]	RAM_ADDR[15]			N11
B[0]	RAM_ADDR[16]			P13
B[1]	RAM_ADDR[17]			Q14
B[8]	RAM_DATA[0]			N13
B[9]	RAM_DATA[1]			P14
B[10]	RAM_DATA[2]			P15
B[11]	RAM_DATA[3]			M14
B[12]	RAM_DATA[4]			L13
B[13]	RAM_DATA[5]			N15
B[14]	RAM_DATA[6]			L14
B[15]	RAM_DATA[7]			M15
C[0]	RAM_DATA[8]	MKPD3_RX_A	SPI1_SS	K13
C[1]	RAM_DATA[9]	MKPD3_nRX_A	SPI1_SCK	K14
C[2]	RAM_DATA[10]	MKPD3_TX_A	SPI1_MISO	L15
C[3]	RAM_DATA[11]	MKPD3_nTX_A	SPI1_MOSI	J14
C[4]	RAM_DATA[12]	MKPD3_TXEN_A	SPI2_SS	J13
C[5]	RAM_DATA[13]	MKPD3_RX_B	SPI2_SCK	K15
C[6]	RAM_DATA[14]	MKPD3_nRX_B	SPI2_MISO	J15
C[7]	RAM_DATA[15]	MKPD3_TXEN_B	SPI2_MOSI	H14
C[8]	RAM_DATA_CHK[0]			G15
C[9]	RAM_DATA_CHK[1]			F15
C[10]	RAM_DATA_CHK[2]			G14
C[11]	RAM_DATA_CHK[3]			F14
C[12]	RAM_DATA_CHK[4]			F13
C[13]	RAM_DATA_CHK[5]		UART1_TxD	E15
C[14]	RAM_DATA_CHK[6]		UART1_RxD	E14
C[15]	RAM_DATA_CHK[7]		UART2_TxD	D15
D[0]	RAM_DATA_CHK[8]		UART2_RxD	C15

Ине. № подл.	Подп. и дата.
Взам. инв. №	Ине. № дубл.
Подп. и дата	

Изм	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 16
-----	------	----------	-------	------	---	------------

Продолжение таблицы 1.2.2

1	2	3	4	5
D[1]	RAM_DATA_CHK[9]			D14
D[2]	RAM_Wen			E13
D[4]	RAM_Oen			C14
D[5]	RAM_LBn			B15
D[6]	RAM_Ubn			D13
D[7]	SPI1_SS	UART1_TxD		C13
D[8]	SPI1_SCK	UART1_RxD		B14
D[9]	SPI1_MISO	UART2_TxD		A14
D[10]	SPI1_MOSI	UART2_RxD		B12
D[11]	SPI2_SS	SW1_SIN		C11
D[12]	SPI2_SCK	SW1_DIN		A13
D[13]	SPI2_MISO	SW1_SOUT		B11
D[14]	SPI2_MOSI	SW1_DOUT		A12
D[15]	UART1_TxD	SW2_SIN		A7
E[0]	UART1_RxD	SW2_DIN		B4
E[1]	UART2_TX	SW2_SOUT		C5
E[2]	UART2_RX	SW2_DOUT		C1
E[3]	RAM_Cen0			E2
E[4]	MKPD1_RX_A	SPI1_SS		C10
E[5]	MKPD1_nRX_A	SPI1_SCK		B10
E[6]	MKPD1_TX_A	SPI1_MISO		A11
E[7]	MKPD1_nTX_A	SPI1_MOSI		B9
E[8]	MKPD1_TXEN_A	SPI2_SS		C9
E[9]	MKPD1_RX_B	SPI2_SCK		A10
E[10]	MKPD1_nRX_B	SPI2_MISO		A9
E[11]	MKPD1_TXEN_B	SPI2_MOSI		A6
E[12]	MKPD2_RX_A		I2C_SCL	B7
E[13]	MKPD2_nRX_A		I2C_SDA	B6
E[14]	MKPD2_TX_A	CAN1_TX		C6

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.

Изм	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 17
-----	------	----------	-------	------	---	------------

Продолжение таблицы 1.2.2

1	2	3	4	5
E[15]	MKPD2_nTX_A	CAN1_RX		A5
F[0]	MKPD2_TXEN_A	CAN2_TX		B5
F[1]	MKPD2_RX_B	CAN2_RX		A4
F[2]	MKPD2_nRX_B	UART3_TxD		A3
F[3]	MKPD2_TXEN_B	UART3_RxD		B3
F[4]	MKPD3_RX_A		UART5_TxD	A2
F[5]	MKPD3_nRX_A		UART5_RxD	C4
F[6]	MKPD3_TX_A		UART6_TxD	C3
F[7]	MKPD3_nTX_A		SW1_SIN	B2
F[8]	MKPD3_TXEN_A		SW1_DIN	B1
F[9]	MKPD3_RX_B		SW1_SOUT	D2
F[10]	MKPD3_nRX_B		SW1_DOUT	E3
F[11]	MKPD3_TXEN_B	RAM_ADDR[23]		D1
F[12]	MKPD4_RX_A	RAM_Cen1		F3
F[13]	MKPD4_nRX_A	RAM_Cen2		F2
F[14]	MKPD4_TX_A	RAM_Cen3		E1
F[15]	MKPD4_nTX_A	RAM_ADDR[18]		G2
G[0]	MKPD4_TXEN_A	RAM_ADDR[19]		G3
G[1]	MKPD4_RX_B	RAM_ADDR[20]		F1
G[2]	MKPD4_nRX_B	RAM_ADDR[21]		G1
G[3]	MKPD4_TXEN_B	RAM_ADDR[22]		K1
G[4]	SW1_SIN	MKPD4_RX_A		J2
G[5]	SW1_DIN	MKPD4_nRX_A		K2
G[6]	SW1_SOUT	MKPD4_TX_A		K3
G[7]	SW1_DOUT	MKPD4_nTX_A		L1
G[8]	SW2_SIN	MKPD4_TXEN_A		L2
G[9]	SW2_DIN	MKPD4_RX_B		M1
G[10]	SW2_SOUT	MKPD4_nRX_B		N1
G[11]	SW2_DOUT	MKPD4_TXEN_B		M2

Ине. № подл.	Подп. и дата	Взам. инв. №	Ине. № дубл.	Подп. и дата.

Изм	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 18
-----	------	----------	-------	------	---	------------

Продолжение таблицы 1.2.2

1	2	3	4	5
G[12]	TM_CLKI		CAN1_TX	L3
G[13]	TM_DOUT_VALID		CAN1_RX	N2
G[14]	TM_DOUT		CAN2_TX	P1
G[15]	TC_CLKI		CAN2_RX	M3
H[0]	TC_ACT		UART3_TxD	N3
H[1]	TC_DIN[0]		UART3_RxD	P2
H[2]	TC_DIN[1]		UART4_TxD	H2
H[3]	TC_DIN[2]		UART4_RxD	J1
H[4]	SW_CLK		UART6_RxD	B8

1.3 Электрические параметры микросхемы

Электрические параметры СБИС CPU при приёмке и поставке должны соответствовать нормам, приведённым в таблице 1.3.1.

Электрические параметры микросхем в процессе и после воздействия специальных факторов (ВСФ) должны соответствовать нормам, приведённым в таблице 1.3.1 для крайних значений рабочей температуры среды.

Значения предельно-допустимых и предельных режимов эксплуатации в диапазоне рабочих температур среды должны соответствовать нормам, приведённым в таблице 1.3.2.

Таблица 1.3.1 — Электрические параметры СБИС CPU при приёмке и поставке

Наименование параметра, единица измерения, режим измерения	Буквенное обозначение параметра	Норма параметра				Температура среды, °С
		До ВСФ		Во время и после ВСФ		
		не менее	не более	не менее	не более	
1	2	3	4	5	6	7
1 Выходное напряжение низкого уровня буферов ввода-вывода, В, при $U_{CC1} = 3,0$ В; $U_{CC2} = 1,62$ В, $I_{OL} \leq 2,0$ мА	U_{OL}	—	0,4	—	0,4	25 ± 10
			0,5		0,5	от – 60 до + 85

Ине. № подл.	Ине. № дубл.	Подп. и дата.
Взам. инв. №	Ине. № дубл.	Подп. и дата.
Подп. и дата		
Ине. № подл.		

Изм	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 19
-----	------	----------	-------	------	---	------------

Продолжение таблицы 1.3.1

1	2	3	4	5	6	7
2 Выходное напряжение высокого уровня буферов ввода-вывода, В, при $U_{CC1} = 3,0 \text{ В}$; $U_{CC2} = 1,62 \text{ В}$, $I_{OH} \leq -0,3 \text{ мА}$	U_{OH}	2,7	–	2,0	–	25 ± 10
		2,3		1,8		от – 60 до + 85
3 Ток потребления в статическом режиме, мА, при $U_{CC1} = 3,6 \text{ В}$; $U_{CC2} = 1,98 \text{ В}$	I_{CC1}	–	150	–	220	25 ± 10
			250		300	от – 60 до + 85
	I_{CC2}	–	65	–	220	25 ± 10
			70		300	от – 60 до + 85
4 Ток утечки низкого и высокого уровня на входе, мкА, Входы «pulldown» (K3, N1) Входы «pullup» (L1, M2) Вход X2/CLKIN (Q2) Все остальные входы при $U_{CC1} = 3,6 \text{ В}$; $U_{CC2} = 1,98 \text{ В}$, $U_{IL} = 0 \text{ В}$ ($U_{IH} = 3,6 \text{ В}$)	I_{ILL}	– 15	200	– 15	200	25 ± 10
		– 200	15	– 200	15	
		– 55	55	– 55	55	
		– 15	15	– 15	15	
	I_{ILH}	– 25	300	– 25	300	от – 60 до + 85
		– 300	25	– 300	25	
		– 75	75	– 75	75	
		– 25	25	– 25	25	
5 Выходной ток низкого и высокого уровня буфера с третьим состоянием в состоянии «Выключено», мкА, при $U_{CC1} = 3,6 \text{ В}$; $U_{CC2} = 1,98 \text{ В}$ $U_{OL} = 0 \text{ В}$ ($U_{OH} = 3,6 \text{ В}$)	I_{OZL}	– 15	15	– 15	15	25 ± 10
	I_{OZH}	– 25	25	– 25	25	от – 60 до + 85
6 Динамический ток потребления ядра, мА, при $U_{CC1} = 3,6 \text{ В}$; $U_{CC2} = 1,98 \text{ В}$ $f_c = 40 \text{ МГц}$ и $C_L \leq 20 \text{ пФ}^{1)}$	I_{OCC}	–	300	–	450	25 ± 10
			350		550	от – 60 до + 85
7 Функциональный контроль (ФК) при $U_{CC1} = 2,97 \text{ В}$; $U_{CC2} = 1,62 \text{ В}$, при $U_{CC1} = 3,63 \text{ В}$; $U_{CC2} = 1,98 \text{ В}$, $f_c = 50 \text{ МГц}$ и $C_L = 20 \text{ пФ}^{1)}$	ФК					25 ± 10
						от – 60 до + 85
8 Входная емкость, пФ	C_I		12		12	25 ± 10
9 Выходная емкость, пФ	C_O		12		12	± 10
10 Емкость входа/выхода, пФ	C_{IO}		14		14	25 ± 10
1) С учётом всех паразитных ёмкостей.						

Ине. № подл.	Подп. и дата.
Взам. инв. №	Инв. № дубл.
Подп. и дата	

Изм	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 20
-----	------	----------	-------	------	---	------------

Примечание — В процессе и непосредственно после воздействия специального фактора 7.И с характеристикой 7.Иб требования к значениям электрических параметров не предъявляют на время потери работоспособности, указанной в пункте 2.6.1 АЕЯР.431290.994 ТУ.

Таблица 1.3.2 — Предельно-допустимые и предельные режимы эксплуатации микросхемы

Наименование параметра режима, единица измерения	Буквенное обозначение параметра	Предельно-допустимый режим		Предельный режим	
		не менее	не более	не менее	не более
Напряжение питания буферов ввода-вывода, В	U_{CC1}	3,0	3,6	– 0,3	4,5 ¹⁾
Напряжение питания ядра, В	U_{CC2}	1,62	1,98	– 0,2	2,5 ¹⁾
Входное напряжение низкого уровня на входах, В	U_{IL}	0	0,8	– 0,3	–
Входное напряжение высокого уровня на входах, В	U_{IH}	2,0	U_{CC1}	–	$U_{CC1} + 0,3$
Выходное напряжение на выходе с третьим состоянием в состоянии «Выключено», В	U_{OZ}	0	U_{CC1}	– 0,3	$U_{CC1} + 0,3$
Выходной ток низкого уровня, мА	I_{OL}	–	1,5	–	2,0
Выходной ток высокого уровня, мА	I_{OH}		– 0,3		– 0,5
Частота следования импульсов тактовых сигналов, МГц	f_C	5	50 ⁴⁾	–	–
Время нарастания и спада входных сигналов, нс	t_{LH} и t_{HL}	–	$0,05T_{CLK}^{2)}$	–	$0,05T_{CLK}^{2)}$
Ёмкость нагрузки, пФ	C_L	–	20 ³⁾	–	40 ³⁾
¹⁾ С учётом всех видов помех. ²⁾ При переключении входных сигналов. ³⁾ С учётом всех паразитных емкостей. ⁴⁾ С приёмкой «5». Примечание – Время работы микросхем в одном из предельных режимов по току или напряжению не должно превышать 5 с.					

Име. № подл.	Подп. и дата.
Взам. инв. №	Инв. № дубл.
Подп. и дата	
Име. № подл.	

Изм	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 21
-----	------	----------	-------	------	---	------------

Таблица 1.3.3 — Средний ток потребления ядра в основных режимах работы микросхемы (напряжение питания $U_{cc2} = 1,98 \text{ В}$, $T = 25,0 \text{ }^{\circ}\text{C}$, $f_c = 50 \text{ МГц}$ и $f_c = 80 \text{ МГц}$)

Режимы работы	Ток потребления, мА	
	$f_c = 50 \text{ МГц}$	$f_c = 80 \text{ МГц}$
Работает ПдТМИ и UART1	481	665
Работает ПрТКИ и UART1	301	403
Работает вычислитель с плавающей запятой и UART1	236	302
Работают 4 контроллера по ГОСТ Р 52070 и UART1	315	424
Работают 2 контроллера интерфейса SpaceWire и UART1	321	440
Работают 2 контроллера SPI и UART1	220	284
Работают 2 контроллера UART	217	285

СБИС CPU стойкая к воздействию специальных факторов, приведённых в таблице 1.3.4.

Таблица 1.3.4 — Стойкость к воздействию специальных факторов

Характеристики специальных факторов	Значения характеристик специальных факторов
7.И1	4Ус
7.И6	4Ус
7.И7	4Ус
7.И8	$0,02 \cdot 1Ус$
7.С1	4Ус
7.С4	4Ус
7.К1	$1K^1) (2K^2);$
7.К4	1K
7.К12	$60^3) \text{ МэВ} \cdot \text{см}^2/\text{мг}$
¹⁾ При совместном воздействии факторов 7.К1 и 7.К4. ²⁾ При независимом воздействии факторов 7.К1 и 7.К4. ³⁾ По катастрофическим отказам и тиристорному эффекту.	

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>					Лист
Изм.	Лист	№ докум.	Подп.	Дата						22

2 Общая характеристика СБИС CPU

Системная шина микросхемы разделена на высокоскоростную шину Advanced Microcontroller Bus Architecture (AMBA) High-performance Bus (AHB) и низкоскоростную AMBA Advanced Peripheral Bus (APB).

Шина AMBA AHB предназначена для связи высокоскоростных устройств:

- ядра процессора;
- оперативной памяти;
- контроллера прямого доступа к памяти (ПДП);
- интерфейсов SpaceWire;
- блоков мультиплексного канала передачи данных (МКПД);
- блоков ПдТМИ и ПрТКИ.

Шина AMBA APB предназначена для подключения низкоскоростных устройств:

- контроллеров последовательных портов;
- интерфейсов управления ПдТМИ и ПрТКИ и ПДП;
- таймеров;
- контроллеров интерфейсов CAN;
- I²C.

Функционирование шины обеспечивается блоками:

— AHB BusMUX — мультиплексор ведущих и ведомых устройств шины AMBA AHB. Данный блок выполняет арбитраж доступа ведущих устройств на шине к ведомым, а также передачу ответов в обратную сторону, обеспечивая доступ к шине всем устройствам с одинаковым приоритетом;

— AHB-APB МОСТ выполняет передачу данных между шинами AHB и APB;

— APB SLAVE MUX — мультиплексор ведомых устройств шины AMBA APB. Данный блок выполняет передачу сигналов управления на ведомые устройства шины, а также передачу ответов в обратную сторону.

Блок ФТС выполняет формирования сетки тактовых частот, необходимых для работы схемы, а также пересинхронизацию сигнала сброса.

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	<div>Микросхема интегральная 5023BC016</div> <div>Техническое описание</div> <div>Версия 2.04 от 01.09.2017 г.</div>					Лист
Изм	Лист	№ докум.	Подп.	Дата						23

3 Описание микросхемы

3.1 Структурная организация микросхемы

Структурная схема СБИС CPU приведена на рисунке 3.1.1.

СБИС CPU состоит из следующих функциональных блоков:

- 32-разрядного процессорного ядра (CPU CORE);
- мультиплексора устройств шины AMBA AHB (матрица переключений шины AHB);
- моста между шинами AMBA AHB и AMBA APB (AHB-APB МОСТ);
- мультиплексора ведомых устройств шины AMBA APB (APBSLAVEMUX);
- блока формирования тактовых сигналов и сигналов сброса (ФТС);
- контроллера прямого доступа к памяти (ПДП);
- четырех резервированных контроллеров последовательного интерфейса по ГОСТ Р 52070 (МКПД1 – МКПД4);
- двух контроллеров интерфейса SpaceWire со скоростью передачи данных до 100 Мбит/с (SW1 – SW2);
- внутрикристальной схемы отладки с интерфейсом JTAG (IEEE 1149.1) (BCO);
- блока граничного сканирования (ГС);
- встроенного оперативного запоминающего устройства, состоящего из двух блоков по 64 кБ (ОЗУ1 – ОЗУ2);
- контроллера встроенного ОЗУ с блоком обнаружения и исправления ошибок;
- контроллера внешней статической памяти со встроенным блоком обнаружения и исправления ошибок (КВП);
- шести асинхронных последовательных портов UART (КАПП1 – КАПП6);
- двух последовательных портов SPI (КПП1 – КПП2);
- специализированного вычислителя с плавающей запятой (BC);
- четырех таймеров общего назначения (Т1 – Т4);
- сторожевого таймера (СТ);

Инв. № подл.	Подп. и дата				<div>Микросхема интегральная 5023BC016</div> <div>Техническое описание</div> <div>Версия 2.04 от 01.09.2017 г.</div>	Лист					
	Инв. № дубл.										
	Взам. инв. №										
	Подп. и дата										
Изм.	Подп.										
	Дата										
Изм					Лист						
№ докум.					Подп.						
Дата											

—	внутрикристальной	схемы	отладки	с	интерфейсом	JTAG	(IEEE 1149.1)
(BCO);							
—	блока	граничного	сканирования	(ГС);			
—	встроенного оперативного запоминающего устройства, состоящего из двух						
блоков по 64 кБ (ОЗУ1 – ОЗУ2);							
—	контроллера встроенного ОЗУ с блоком обнаружения и исправления						
ошибок;							
—	контроллера внешней статической памяти со встроенным блоком						
обнаружения и исправления ошибок (КВП);							
—	шести асинхронных последовательных портов UART (КАПП1 – КАПП6);						
—	двух последовательных портов SPI (КПП1 – КПП2);						
—	специализированного вычислителя с плавающей запятой (BC);						
—	четырёх таймеров общего назначения (Т1 – Т4);						
—	сторожевого таймера (СТ);						

Изм	Лист	№ докум.	Подп.	Дата

- контроллера передатчика телеметрической информации в соответствии со стандартом CCSDS (ПдТМИ);
- контроллера приёмника телекомандной информации в соответствии со стандартом CCSDS (ПрТКИ);
- двух контроллеров интерфейса CAN (CAN1 – CAN2);
- одного контроллера интерфейса I²C (ПС Master).

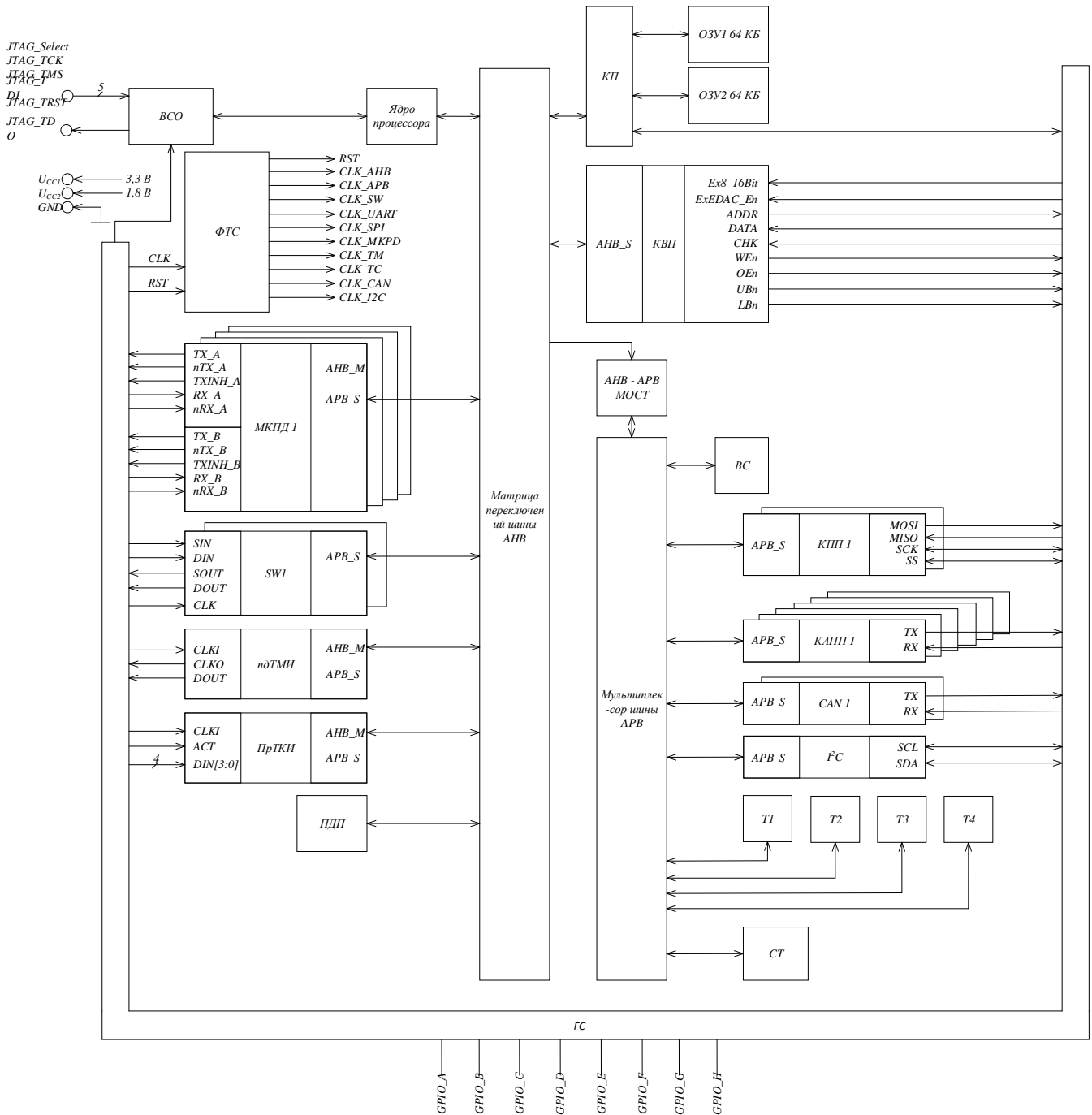


Рисунок 3.1.1 — Структурная схема СБИС CPU

Име. № подл.	Подп. и дата.	Взам. име. №	Име. № дубл.	Подп. и дата.

Изм.	Лист	№ докум.	Подп.	Дата

Микросхема интегральная 5023BC016
Техническое описание
Версия 2.04 от 01.09.2017 г.

3.1.1 Конфигурационные выводы микросхемы

JTAG_SELECT:

— 0 — выводы JTAG перенаправлены к программатору и отладчику ARM процессора;

— 1 — выводы JTAG перенаправлены к Boundary Scan контроллеру.

Ex8_16bit: режим работы внешней памяти по Chip Select 1:

— 0 — 8-битный режим;

— 1 — 16-битный режим.

ExEDAC_En: включение режима исправления единичных и обнаружения неисправимых ошибок во внешней памяти по Chip Select 1.

В 8-битном режиме используются выводы RAM_DATA_CHK [4:0], в 16-битном режиме — RAM_DATA_CHK [9:0].

3.1.2 Инициализация работы микросхемы

Перед началом работы микросхемы должны быть настроены внешние выводы контроллера внешней памяти ExEDAC_En и Ex8_16bit. ExEDAC_En управляет включением/выключением блока обнаружения и исправления ошибок (EDAC) контроллера внешней памяти по Chip Select 1.

Начальная загрузка происходит через Chip Select 1. В таблице 3.1.1 представлено соответствие Chip Select и выводов.

Таблица 3.1.1 — Соответствие Chip Select и выводов

CS	Название функции	Номер порта ввыда-вывода	Номер альтернативной функции	Номер вывода
Chip Select 1	RAM_Cen0	E[3]	ALT_FUNC_1	E2
Chip Select 2	RAM_Cen1	F[12]	ALT_FUNC_2	F3
Chip Select 3	RAM_Cen2	F[13]	ALT_FUNC_2	F2
Chip Select 4	RAM_Cen3	F[14]	ALT_FUNC_2	E1

Если на вывод ExEDAC_En подаётся уровень логической единицы, то EDAC включен. Если ExEDAC_En подключен к земле, то EDAC выключен. Ex8_16bit переключает режим 8/16 бит контроллера внешней памяти. При Ex8_16bit равном

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	<div>Микросхема интегральная 5023BC016</div> <div>Техническое описание</div> <div>Версия 2.04 от 01.09.2017 г.</div> <div>Лист 26</div>				
Изм.	Лист	№ докум.	Подп.	Дата					

логической единице контроллер внешней памяти находится в 16-битном режиме, в обратном случае — в 8-битном.

После подачи питаний U_{CC1} и U_{CC2} микросхема должна быть сброшена внешним сигналом на выводе RST. Одновременно с сигналом сброса должен подаваться тактовый сигнал на вывод CLK. Минимальная длительность сигнала сброса должна быть не менее 2 периодов тактового сигнала. Активный уровень сигнала сброс — логический ноль.

Перед началом работы с периферийными модулями микросхемы необходимо включить тактовые сигналы этих модулей. Включение тактовых сигналов осуществляется в регистре общего назначения PWR_CTRL_CLK. Также должен быть снят сброс соответствующего модуля в регистре PWR_CTRL_RST.

3.2 Распределение адресного пространства СБИС CPU

Распределение адресного пространства микросхемы показано в таблице 3.2.1.

Таблица 3.2.1 — Структура адресного пространства

Адрес	Описание
1	2
0x0000_0000 – 0x00FF_FFFF	Алиасинг для областей: Внешняя память Chip Select 1-4, внутренняя память блок 1, внутренняя память блок 2. По сбросу: внешняя память Chip Select 1
0x0800_0000 – 0x08FF_FFFF	Внешняя память Chip Select 1
0x0900_0000 – 0x09FF_FFFF	Внешняя память Chip Select 2
0x0A00_0000 – 0x0AFF_FFFF	Внешняя память Chip Select 3
0x0B00_0000 – 0x0BFF_FFFF	Внешняя память Chip Select 4
0x2000_0000 – 0x2000_FFFF	Внутренняя память блок 1
0x2001_0000 – 0x2001_FFFF	Внутренняя память блок 2
0x8000_0000 – 0x8000_0FFF	GPIO_A
0x8001_0000 – 0x8001_0FFF	GPIO_B
0x8002_0000 – 0x8002_0FFF	GPIO_C
0x8003_0000 – 0x8003_0FFF	GPIO_D
0x8004_0000 – 0x8004_0FFF	GPIO_E
0x8005_0000 – 0x8005_0FFF	GPIO_F
0x8006_0000 – 0x8006_0FFF	GPIO_G

Ине. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.

Продолжение таблицы 3.2.1

1	2
0x8007_0000 – 0x8007_0FFF	GPIO_H
0x8008_0000 – 0x8008_0FFF	GPIO_I
0xA000_0000 – 0xA000_FFFF	Регистры общего назначения
0xA001_0000 – 0xA001_0FFF	DMA контроллер
0xA002_0000 – 0xA002_0FFF	SPI_1
0xA003_0000 – 0xA003_0FFF	SPI_2
0xA004_0000 – 0xA004_0FFF	UART_1
0xA005_0000 – 0xA005_0FFF	UART_2
0xA006_0000 – 0xA006_0FFF	UART_3
0xA007_0000 – 0xA007_0FFF	UART_4
0xA008_0000 – 0xA008_0FFF	WatchDog
0xA009_0000 – 0xA009_0FFF	Timer 1
0xA00A_0000 – 0xA00A_FFFF	SpaceWire_1
0xA00B_0000 – 0xA00B_FFFF	SpaceWire_2
0xA00C_0000 – 0xA00C_FFFF	МКПД_1
0xA00D_0000 – 0xA00D_FFFF	МКПД_2
0xA00E_0000 – 0xA00E_FFFF	МКПД_3
0xA00F_0000 – 0xA00F_FFFF	МКПД_4
0xA010_0000 – 0xA010_FFFF	Telemetry Encoder (TMTX)
0xA011_0000 – 0xA011_FFFF	TCRX
0xA012_0000 – 0xA012_FFFF	Floating Point Unit (FPU)
0xA013_0000 – 0xA013_FFFF	Timer 2
0xA014_0000 – 0xA014_FFFF	Timer 3
0xA015_0000 – 0xA015_FFFF	Timer 4
0xA01B_0000 – 0xA01B_FFFF	CAN 1
0xA01C_0000 – 0xA01C_FFFF	CAN 2
0xA01D_0000 – 0xA01D_FFFF	Мультиплексор прерываний
0xA01E_0000 – 0xA01E_FFFF	UART_5
0xA01F_0000 – 0xA01F_FFFF	UART_6
0xA020_0000 – 0xA020_FFFF	I ² C
0x6000_0000 – 0x6000_FFFF	Тестовый доступ к памяти блок 1, данные
0x6010_0000 – 0x6010_FFFF	Тестовый доступ к памяти блок 1, биты ECC (доступно 20 бит, маска 0x1F1F1F1F)

Ине. № подл.	Подп. и дата.
Взам. инв. №	Инв. № дубл.
Подп. и дата	

Изм	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 28
-----	------	----------	-------	------	---	------------

Копировал

Формат А4

Продолжение таблицы 3.2.1

1	2
0x6100_0000 – 0x6100_3FFF	Тестовый доступ к памяти кэша way1 данные
0x6110_0000 – 0x6110_3FFF	Тестовый доступ к памяти кэша way1 таг и сгс (доступен 21 бит)
0x6200_0000 – 0x6200_FFFF	Тестовый доступ к памяти блок 2, данные
0x6210_0000 – 0x6210_FFFF	Тестовый доступ к памяти блок 2, биты ECC (доступно 20 бит)

3.3 Вектора прерываний СБИС CPU

Процессор имеет 32 вектора прерываний.

Управляющие регистры ссылаются на базовый адрес Мультиплексора прерываний — 0xA01D_0000.

Каждому источнику прерывания соответствует свой управляющий регистр, с адресом, равным номеру источника прерываний, умноженному на 4.

Для того чтобы подключить требуемый источник прерываний на определенный вектор прерываний, нужно запрограммировать соответствующий (соответствует номеру источника прерывания) управляющий регистр мультиплексора прерываний. Также необходимо разрешить прерывания, вызвав NVIC_En_IRQ(vec), где vec — номер вектора прерываний.

Источники прерываний, которые подключены на один и тот же вектор прерываний, объединяются по «ИЛИ».

3.3.1 Регистры INTR_MUX_CTRL

Таблица 3.3.1 содержит описание регистра INTR_MUX_CTRL.

Таблица 3.3.1 — Описание регистра INTR_MUX_CTRL

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31	INTR_SRC_ENABLE (RW)	1 — включение источника прерывания
30 – 5	–	
4 – 0	NUM_INTR_VECTOR (RW)	номер вектора прерываний от 0 до 31, на который направляется соответствующий источник прерываний

Име. № подл.	Подп. и дата
Взам. инв. №	Инв. № дубл.
Подп. и дата	Подп. и дата

Изм.	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 29
------	------	----------	-------	------	---	------------

3.3.1.1 Номера источников прерываний

Таблица 3.3.2 — Номера источников прерываний

Номер источника прерываний	Наименование источника прерываний	Номер источника прерываний	Наименование источника прерываний	Номер источника прерываний	Наименование источника прерываний
1	2	1	2	1	2
00	Watchdog	26	UART 3	53	CAN 2
01	Timer 1	27	UART 4	56	EDAC
03	Timer 2	28	UART 5	98	DMA SPI 2 TX
04	Timer 3	29	UART 6	99	DMA SPI 2 RX
08	Timer 4	32	SPI 1	100	DMA UART 1 TX
09	GPIO A	33	SPI 2	101	DMA UART 1 RX
10	GPIO B	36	SpaceWire 1	102	DMA UART 2 TX
11	GPIO C	37	SpaceWire 2	103	DMA UART 2 RX
12	GPIO D	40	МКПД 1	104	DMA UART 3 TX
13	GPIO E	41	МКПД 2	105	DMA UART 3 RX
14	GPIO F	42	МКПД 3	106	DMA UART 4 TX
15	GPIO G	43	МКПД 4	107	DMA UART 4 RX
16	GPIO H	48	TMTX	108	DMA UART 5 TX
23	GPIO I	49	TCRX	109	DMA UART 5 RX
24	UART 1	50	I ² C	110	DMA UART 6 TX
25	UART 2	52	CAN 1	111	DMA UART 6 RX

3.4 Регистры общего назначения

3.4.1 Состав регистров общего назначения

Состав регистров общего назначения приведён в таблице 3.4.1.

Назначения разрядов регистров общего назначения приведены в таблицах 3.4.2 – 3.4.38.

Таблица 3.4.1 — Состав регистров общего назначения

Смещение	Название регистра	Тип	Значение по сбросу	Описание
1	2	3	4	5
0 x 00	EXTMEM_CTRL	RW	0x1FF	Конфигурирование контроллера внешней памяти по Chip Select 1
0 x 04	EDAC_CTRL	RW	0x3	Включение режима исправления и детекции ошибок внутренней памяти.
0 x 08	INTMEM_CERR_CNT	RO	0	Счётчик исправленных ошибок внутренней памяти блока 1.
0 x 0C	INTMEM_FERR_CNT	RO	0	Счётчик неисправимых ошибок внутренней памяти блока 1.

Микросхема интегральная 5023BC016

Техническое описание

Версия 2.04 от 01.09.2017 г.

Лист

30

Копировал

Формат А4

Продолжение таблицы 3.4.1

1	2	3	4	5
0 x 10	EXTMEM_CERR_CNT	RO	0	Счётчик исправленных ошибок внешней памяти.
0 x 14	EXTMEM_FERR_CNT	RO	0	Счётчик неисправимых ошибок внешней памяти.
0 x 1C	SPACEWIRE_CLK_CTRL	RW	0	Выбор тактовой частоты блоков SpaceWire
0 x 20	INTMEM2_CERR_CNT	RO	0	Счётчик исправленных ошибок внутренней памяти блока 2.
0 x 24	INTMEM2_FERR_CNT	RO	0	Счётчик неисправимых ошибок внутренней памяти блока 2.
0 x 28	PWR_CTRL_CLK	RW	0	Включение тактовых частот различных блоков
0 x 2C	PWR_CTRL_RST	RW	0	Асинхронный сброс различных блоков
0 x 30	EXTMEM2_CTRL	RW	0x1FF	Конфигурирование контроллера внешней памяти по Chip Select 2
0 x 34	EXTMEM3_CTRL	RW	0x1FF	Конфигурирование контроллера внешней памяти по Chip Select 3
0 x 38	EXTMEM4_CTRL	RW	0x1FF	Конфигурирование контроллера внешней памяти по Chip Select 4
0 x 3C	CACHE_HIGH_ADDR	RW	0	Задаёт пространство работы кэша
0 x 40	INTMEM_SCR_RNG_ADDR	RW	0x3	Диапазон адресов скраббера для блока памяти 1.
0 x 44	INTMEM_SCR_PRD_SCAN	RW	0	Период сканирования скраббера для блока памяти 1.
0 x 48	INTMEM_SCR_PRD_STOP	RW	0	Период прерывания скраббером контроллер памяти 1.
0 x 4C	INTMEM2_SCR_RNG_ADDR	RW	0x3	Диапазон адресов скраббера для блока памяти 2.
0 x 50	INTMEM2_SCR_PRD_SCAN	RW	0	Период сканирования скраббера для блока памяти 2.
0 x 54	INTMEM2_SCR_PRD_STOP	RW	0	Период прерывания скраббером контроллер памяти 2.
0 x 58	INTMEMS_SCR_MAIN	RW	0	Общее управление скрабберами.
0 x 5C	CACHE_CRC_ERROR	RO	0	Счётчик сгс ошибок в кэше
0 x 60	EDAC_INTMEM_SCR_CERR	RO	0	Счётчик исправимых ошибок, обнаруженных скраббером блока памяти 1.

Ине. № подл.	Подп. и дата	Взам. инв. №	Ине. № дубл.	Подп. и дата.

Изм	Лист	№ докум.	Подп.	Дата

Микросхема интегральная 5023BC016

Техническое описание
Версия 2.04 от 01.09.2017 г.

Лист

31

Копировал

Формат А4

Продолжение таблицы 3.4.1

1	2	3	4	5
0 x 64	EDAC_INTMEM_SCR_FERR	RO	0	Счётчик неисправимых ошибок, обнаруженных скраббером блока памяти 1.
0 x 68	EDAC_INTMEM2_SCR_CERR	RO	0	Счётчик исправимых ошибок, обнаруженных скраббером блока памяти 2.
0 x 6C	EDAC_INTMEM2_SCR_FERR	RO	0	Счётчик неисправимых ошибок, обнаруженных скраббером блока памяти 2.
0 x 70	DMA_INTR_FLAGS	RW1C	0	Флаги прерываний Direct Memory Access (DMA)
0 x 74 – 0 x 94	ALT_FUNCTION_CTRL	RW	0	Выбор альтернативной функции для каждого порта GPIO
0 x A0	CACHE_HIGH_ADDR_CS2	RW	0	Задаёт пространство работы кэша для памяти по Chip Select 2
0 x A4	CACHE_HIGH_ADDR_CS3	RW	0	Задаёт пространство работы кэша для памяти по Chip Select 3
0 x A8	CACHE_HIGH_ADDR_CS4	RW	0	Задаёт пространство работы кэша для памяти по Chip Select 4
0 x AC	ALIAS_CTRL	RW	0	Настройка алиасинга для области адресов 0x0000_0000-0x00FF_FFFF
0 x B0	SCRUBBER_FERR_ADDR	RO	0	Адрес последнего FERR для скрабберов.
0 x B4	COMMON_FERR_ADDR	RO	0	Адрес последнего FERR для внешней и внутренней памяти без скраббера.
0 x B8	EDAC_REACTION_CTRL	RW	0	Управление реакцией на ошибки EDAC и CRC
0 x BC	GLOBAL_RESET	RW	0	Сброс всех блоков, включая отладчик (W1O)
0 x C4	CACHE_MAIN	RW	0	Включение / сброс и статус кэша
0 x C8 – 0 xFFFF	—			Не используется, читается 0.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.

Изм	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 32
-----	------	----------	-------	------	---	------------

Копировал

Формат А4

3.4.1.1 Регистр EXTMEM_CTRL

Таблица 3.4.2 — Описание регистра EXTMEM_CTRL

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31	EDAC_FORCE	Если поле имеет значение «1», использование блока исправления ошибок по Chip Select 1 определяется полем EDAC, иначе определяется значением внешнего вывода EXTMEM_EDAC_EN.
30	SIZE_FORCE	Если поле имеет значение «1», разрядность шины данных внешней памяти по Chip Select 1 определяется полем Size, иначе определяется значением внешнего вывода EXTMEM_CFGSIZE.
29 – 25	–	
24	EDAC	Включение блока исправления ошибок внешней памяти по Chip Select 1. Значение 0 — блок исправления ошибок не используется, 1 — блок исправления ошибок включен. Имеет значение, только если бит EDAC_FORCE выставлен в «1».
23 – 17	–	
16	SIZE	Выбор режима разрядности внешней памяти по Chip Select 1. Значение 0 — 8-битная шина данных, значение 1 — 16-битная шина данных. Имеет значение, только если бит SIZE_FORCE выставлен в «1».
15 – 9	–	
8 – 6	TURN_CYCLES	Величина задержки Chip Select (сигнал CEN) в тактах системной частоты, которая приводит к паузе перед каждой операцией записи или чтения. Значение 0 — нет задержки, значение 1 — 1 такт задержки, значение 7 — 7 тактов задержки.
5 – 3	WRITE_CYCLES	Количество тактов системной частоты, за время которых происходит операция записи во внешнюю память. Значение 0 — 1 такт, значение 1 — 2 такта, значение 7 — 8 тактов.
2 – 0	READ_CYCLES	Количество тактов системной частоты, за время которых происходит операция чтения из внешней памяти. Значение 0 — 1 такт, значение 1 — 2 такта, значение 7 — 8 тактов.

3.4.1.2 Регистр EDAC_CTRL

Таблица 3.4.3 — Описание регистра EDAC_CTRL

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 4	–	
3	EDAC_WR_DIS_2	Отключение записи битов EDAC для блока памяти 2.
2	EDAC_WR_DIS_1	Отключение записи битов EDAC для блока памяти 1.
1	EDAC_En_2	Включение режима исправления единичных и обнаружения неисправимых ошибок во внутренней памяти блока 2.
0	EDAC_En_1	Включение режима исправления единичных и обнаружения неисправимых ошибок во внутренней памяти блока 1.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	Микросхема интегральная 5023BC016					Лист
					Техническое описание					
					Версия 2.04 от 01.09.2017 г.					33
					Изм	Лист	№ докум.	Подп.	Дата	

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 0	INTMEM_CERR_CNT	Счётчик исправленных единичных ошибок внутренней памяти блока 1.

Таблица 3.4.5 — Описание регистра INTMEM_FERR_CNT

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 0	INTMEM_FERR_CNT	Счётчик неисправимых ошибок внутренней памяти блока 1.

Таблица 3.4.6 — Описание регистра EXTMEM_CERR_CNT

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 0	EXTMEM_CERR_CNT	Счётчик исправленных единичных ошибок внешней памяти.

Таблица 3.4.7 — Описание регистра EXTMEM_FERR_CNT

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 0	EXTMEM_FERR_CNT	Счётчик неисправимых ошибок внешней памяти.

Таблица 3.4.8 — Описание регистра SPACEWIRE_CLK_CTRL

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 17	–	
16	SPACEWIRE_CLK	Выбор тактовой частоты физического уровня блоков SpaceWire. 0 — тактирование осуществляется с внешнего вывода SW_CLK (PORTH[4]). 1 — тактирование осуществляется системной частотой.
15 – 0	–	

3.4.1.8 Регистр INTMEM2_CERR_CNT

Таблица 3.4.9 — Описание регистра INTMEM2_CERR_CNT

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 0	NTMEM2_CERR_CNT	Счётчик исправленных единичных ошибок внутренней памяти блока 2.

3.4.1.9 Регистр INTMEM2_FERR_CNT

Таблица 3.4.10 — Описание регистра INTMEM2_FERR_CNT

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 0	INTMEM2_FERR_CNT	Счётчик неисправимых ошибок внутренней памяти блока 2.

3.4.1.10 Регистр PWR_CTRL_CLK

Таблица 3.4.11 — Описание регистра PWR_CTRL_CLK

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 0	PWR_CTRL_CLK	Включение тактовых частот для различных блоков. (1 — вкл).

3.4.1.11 Регистр PWR_CTRL_RST

Таблица 3.4.12 — Описание регистра PWR_CTRL_RST

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 0	PWR_CTRL_RST	Включение путем вывода из асинхронного сброса для различных блоков. (1 — вкл).

Расшифровка групп битовых полей для регистров PWR_CTRL_CLK и PWR_CTRL_RST представлена в таблице 3.4.13. Каждый бит соответствует включению одного модуля.

Таблица 3.4.13 — Расшифровка групп битовых полей для регистров PWR_CTRL_CLK и PWR_CTRL_RST.

Номер источника прерываний	Наименование источника прерываний	Номер источника прерываний	Наименование источника прерываний
1	2	1	2
0	SPI_1	11	МКПД_4
1	SPI_2	12	Telemetry Encoder (TMTX)
2	UART_1	13	TCRX
3	UART_2	14	FPU

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.					Лист
					Изм	Лист	№ докум.	Подп.	Дата	
										35

Продолжение таблицы 3.4.13

1	2	1	2
4	UART_3	20	CAN_1
5	UART_4	21	CAN_2
6	SpaceWire_1	22	UART_5
7	SpaceWire_2	23	UART_6
8	МКПД_1	24	I ² C
9	МКПД_2	25 – 31	не используются
10	МКПД_3		

3.4.1.12 Регистр EXTMEM2_CTRL (Chip Select 2)

Таблица 3.4.14 — Описание регистра EXTMEM2_CTRL

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 25	–	
24	EDAC_2	для внешней памяти: 0 — выключен; 1 — включен
23 – 17	–	
16	SIZE_2	0 — 8-битный режим; 1 — 16-битный режим
15 – 9	–	
8 – 6	TURN_CYCLES_2	величина задержки Chip Select (сигнал CEN) в тактах системной частоты, которая приводит к паузе перед каждой операцией записи или чтения. Значение 0 — нет задержки, значение 1 — 1 такт задержки, ..., значение 7 — 7 тактов задержки
5 – 3	WRITE_CYCLES_2	количество системных тактов, за время которых происходит операция записи во внешнюю память. Значение 0 — 1 такт, значение 1 — 2 такта, ... , значение 7 — 8 тактов
2 – 0	READ_CYCLES_2	количество системных тактов, за время которых происходит операция чтения из внешней памяти. Значение 0 — 1 такт, значение 1 — 2 такта, ... , значение 7 — 8 тактов

3.4.1.13 Регистр EXTMEM3_CTRL (Chip Select 3)

Таблица 3.4.15 — Описание регистра EXTMEM3_CTRL

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
1	2	3
31 – 25	–	
24	EDAC_3	для внешней памяти: 0 — выключен; 1 — включен
23 – 17	–	
16	SIZE_3	0 — 8-битный режим; 1 — 16-битный режим
15 – 9	–	

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>					Лист
Изм.	Лист	№ докум.	Подп.	Дата						36

Продолжение таблицы 3.4.15

1	2	3
8 – 6	TURN_CYCLES_3	величина задержки Chip Select (сигнал CEN) в тактах системной частоты, которая приводит к паузе перед каждой операцией записи или чтения. Значение 0 — нет задержки, значение 1 — 1 такт задержки, ..., значение 7 — 7 тактов задержки
5 – 3	WRITE_CYCLES_3	количество системных тактов, за время которых происходит операция записи во внешнюю память. Значение 0 — 1 такт, значение 1 — 2 такта, ... , значение 7 — 8 тактов
2 – 0	READ_CYCLES_3	количество системных тактов, за время которых происходит операция чтения из внешней памяти. Значение 0 — 1 такт, значение 1 — 2 такта, ... , значение 7 — 8 тактов

3.4.1.14 Регистр EXTMEM4_CTRL (Chip Select 4)

Таблица 3.4.16 — Описание регистра EXTMEM4_CTRL

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 25	–	
24	EDAC_4	для внешней памяти: 0 — выключен; 1 — включен
23 – 17	–	
16	SIZE_4	0 — 8-битный режим; 1 — 16-битный режим
15 – 9	–	
8 – 6	TURN_CYCLES_4	величина задержки Chip Select (сигнал CEN) в тактах системной частоты, которая приводит к паузе перед каждой операцией записи или чтения. Значение 0 — нет задержки, значение 1 — 1 такт задержки, ..., значение 7 — 7 тактов задержки
5 – 3	WRITE_CYCLES_4	количество системных тактов, за время которых происходит операция записи во внешнюю память. Значение 0 — 1 такт, значение 1 — 2 такта, ... , значение 7 — 8 тактов
2 – 0	READ_CYCLES_4	количество системных тактов, за время которых происходит операция чтения из внешней памяти. Значение 0 — 1 такт, значение 1 — 2 такта, ... , значение 7 — 8 тактов

3.4.1.15 Регистр CACHE_HIGH_ADDR

Таблица 3.4.17 — Описание регистра CACHE_HIGH_ADDR

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 24	–	
23 – 2	CACHE_HIGH_ADDR	задает старший адрес в памяти, используемый блоком кэширования. Кэшироваться будут данные, расположенные по адресам между 0 и указанным значением не включительно. При увеличении значения этого поля, происходит переинициализация блока кэширования, при уменьшении значения поля переинициализация не происходит
1 – 0	–	

Име. № подл.	Подп. и дата.
Взам. инв. №	Инв. № дубл.
Подп. и дата	
Име. № подл.	

Изм	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 37
-----	------	----------	-------	------	---	------------

3.4.1.16 Регистры CACHE_HIGH_ADDR_CS2, CACHE_HIGH_ADDR_CS3, CACHE_HIGH_ADDR_CS4

Регистры задают старший адрес области кэширования для Chip Select 2, Chip Select 3 и Chip Select 4. Структура регистров описана в таблицах 3.4.18, 3.4.19, 3.4.20.

Таблица 3.4.18 — Описание регистра CACHE_HIGH_ADDR_CS2

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 24	–	
23 – 2	CACHE_HIGH_ADDR2	задаёт старший адрес в памяти, используемый блоком кэширования. Кэшироваться будут данные, расположенные по адресам между 0 и указанным значением не включительно. При увеличении значения этого поля, происходит переинициализация блока кэширования, при уменьшении значения поля переинициализация не происходит
1 – 0	–	

Таблица 3.4.19 — Описание регистра CACHE_HIGH_ADDR_CS3

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 24	–	
23 – 2	CACHE_HIGH_ADDR3	задаёт старший адрес в памяти, используемый блоком кэширования. Кэшироваться будут данные, расположенные по адресам между 0 и указанным значением не включительно. При увеличении значения этого поля, происходит переинициализация блока кэширования, при уменьшении значения поля переинициализация не происходит
1 – 0	–	

Таблица 3.4.20 — Описание регистра CACHE_HIGH_ADDR_CS4

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 24	–	
23 – 2	CACHE_HIGH_ADDR4	задаёт старший адрес в памяти, используемый блоком кэширования. Кэшироваться будут данные, расположенные по адресам между 0 и указанным значением не включительно. При увеличении значения этого поля, происходит переинициализация блока кэширования, при уменьшении значения поля переинициализация не происходит
1 – 0	–	

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.

Изм	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 38
-----	------	----------	-------	------	---	------------

3.4.1.17 Регистры INTMEM_SCR_RNG_ADDR, INTMEM2_SCR_RNG_ADDR

Таблица 3.4.21 — Описание регистров INTMEM_SCR_RNG_ADDR, INTMEM2_SCR_RNG_ADDR

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 16	–	
15 – 3	INTMEM_SCR_RNG_ADDR	задаёт верхнюю границу работы скраббера внутри блока памяти
2 – 0	–	

3.4.1.18 Регистры INTMEM_SCR_PRD_SCAN, INTMEM2_SCR_PRD_SCAN

Таблица 3.4.22 — Описание регистров INTMEM_SCR_PRD_SCAN, INTMEM2_SCR_PRD_SCAN

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 0	INTMEM_SCR_PRD_SCAN	период сканирования заданной области скраббером в тактах. 0 — сканирует постоянно.

3.4.1.19 Регистры INTMEM_SCR_PRD_STOP, INTMEM2_SCR_PRD_STOP

Таблица 3.4.23 — Описание регистров INTMEM_SCR_PRD_STOP, INTMEM2_SCR_PRD_STOP

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 0	INTMEM_SCR_PRD_STOP	период прерывания скраббером контроллера памяти, если в контроллере нет пустых транзакций к соответствующему чипу памяти. 0 — прерывает сразу.

3.4.1.20 Регистр INTMEMS_SCR_MAIN

Таблица 3.4.24 — Описание регистра INTMEMS_SCR_MAIN

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 12	–	
11 – 8	scrub_blk2_cs_ready (RO)	отдельные биты готовности скрабберов по каждому Chip Select (четные и нечетные)
7 – 4	scrub_blk1_cs_ready (RO)	
3	scrub_blk2_ready (RO)	1 — сообщает о готовности всех скрабберов для соответствующего блока памяти
2	scrub_blk1_ready (RO)	
1	scrub_blk2_en (RW)	1 — включает скраббер для соответствующего блока памяти
0	scrub_blk1_en (RW)	

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.					Лист
					Изм.	Лист	№ докум.	Подп.	Дата	
										39

При включении скраббера, первый цикл скраббер не сообщает об исправленных ошибках, считается циклом инициализации не тронутых ячеек памяти, по завершении процедуры выставляется бит в «1». Также после изменения пользователем верхней границы INTMEM[1,2]_SCR_RNG_ADDR в сторону увеличения, производится цикл инициализации, во время которого бит ready равен «0».

3.4.1.21 Регистры CACHE_CRC_ERROR, EDAC_INTMEM[1,2]_SCR_CERR, EDAC_INTMEM[1,2]_SCR_FERR

Таблица 3.4.25 — Описание регистров CACHE_CRC_ERROR, EDAC_INTMEM[1,2]_SCR_CERR, EDAC_INTMEM[1,2]_SCR_FERR

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 0	Counter (RO)	Счётчики ошибок

3.4.1.22 Регистр DMA_INTR_FLAGS

Таблица 3.4.26 — Описание регистра DMA_INTR_FLAGS

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений																		
31 – 8	—																			
7 – 0	DMA_INTR_FLAGS	<p>[15:0] флаги прерываний DMA, сбрасываются записью 1.</p> <table> <tr> <td>[0] — SPI1 TX</td><td>[6] — UART2 TX</td><td>[12] — UART5 TX</td></tr> <tr> <td>[1] — SPI1 RX</td><td>[7] — UART2 RX</td><td>[13] — UART5 RX</td></tr> <tr> <td>[2] — SPI2 TX</td><td>[8] — UART3 TX</td><td>[14] — UART6 TX</td></tr> <tr> <td>[3] — SPI2 RX</td><td>[9] — UART3 RX</td><td>[15] — UART6 RX</td></tr> <tr> <td>[4] — UART1 TX</td><td>[10] — UART4 TX</td><td></td></tr> <tr> <td>[5] — UART1 RX</td><td>[11] — UART4 RX</td><td></td></tr> </table>	[0] — SPI1 TX	[6] — UART2 TX	[12] — UART5 TX	[1] — SPI1 RX	[7] — UART2 RX	[13] — UART5 RX	[2] — SPI2 TX	[8] — UART3 TX	[14] — UART6 TX	[3] — SPI2 RX	[9] — UART3 RX	[15] — UART6 RX	[4] — UART1 TX	[10] — UART4 TX		[5] — UART1 RX	[11] — UART4 RX	
[0] — SPI1 TX	[6] — UART2 TX	[12] — UART5 TX																		
[1] — SPI1 RX	[7] — UART2 RX	[13] — UART5 RX																		
[2] — SPI2 TX	[8] — UART3 TX	[14] — UART6 TX																		
[3] — SPI2 RX	[9] — UART3 RX	[15] — UART6 RX																		
[4] — UART1 TX	[10] — UART4 TX																			
[5] — UART1 RX	[11] — UART4 RX																			

3.4.1.23 Регистры ALT_FUNCTION_CTRL

Для каждого порта ввода-вывода используется отдельный регистр выбора используемой альтернативной функции, всего существует 9 регистров (по количеству портов).

Для каждого вывода микросхемы (например, D[0], D[1]...D[15]) используются 2 бита для выбора номера альтернативной функции. Например, для настройки вывода микросхемы D[7] в режиме ALT_FUNC_2, необходимо записать «1» в 14 бит и «0» в 15 бит. В таблицах 3.4.27 – 3.4.35 приведено описание регистров ALT_FUNCTION_CTRL.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений									
					31 – 8									
					—									
					7 – 0									
					DMA_INTR_FLAGS					[15:0] флаги прерываний DMA, сбрасываются записью 1. [0] — SPI1 TX [6] — UART2 TX [12] — UART5 TX [1] — SPI1 RX [7] — UART2 RX [13] — UART5 RX [2] — SPI2 TX [8] — UART3 TX [14] — UART6 TX [3] — SPI2 RX [9] — UART3 RX [15] — UART6 RX [4] — UART1 TX [10] — UART4 TX [5] — UART1 RX [11] — UART4 RX				

3.4.1.23 Регистры ALT_FUNCTION_CTRL

Для каждого порта ввода-вывода используется отдельный регистр выбора используемой альтернативной функции, всего существует 9 регистров (по количеству портов).

Для каждого вывода микросхемы (например, D[0], D[1]...D[15]) используются 2 бита для выбора номера альтернативной функции. Например, для настройки вывода микросхемы D[7] в режиме ALT_FUNC_2, необходимо записать «1» в 14 бит и «0» в 15 бит. В таблицах 3.4.27 – 3.4.35 приведено описание регистров ALT_FUNCTION_CTRL.

					Микросхема интегральная 5023BC016					Лист
					Техническое описание					
Изм.	Лист	№ докум.	Подп.	Дата	Версия 2.04 от 01.09.2017 г.					40

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
от $N \cdot 2$ до $N \cdot 2 + 1$	BIT_FUNC	Для бита N ($N=0 \dots 15$, номер вывода микросхемы) значение поля определяет используемую альтернативную функцию: 0 — функция 1; 2 — функция 3; 1 — функция 2; 3 — не используется.

Таблица 3.4.29 — Описание регистра ALT_FUNCTION_CTRL_C

Таблица 3.4.30 — Описание регистра ALT_FUNCTION_CTRL_D

Таблица 3.4.31 — Описание регистра ALT_FUNCTION_CTRL_E

Таблица 3.4.32 — Описание регистра ALT_FUNCTION_CTRL_F

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
от $N \cdot 2$ до $N \cdot 2 + 1$	BIT_FUNC	Для бита N ($N=0 \dots 15$, номер вывода микросхемы) значение поля определяет используемую альтернативную функцию: 0 — функция 1; 2 — функция 3; 1 — функция 2; 3 — не используется.

Таблица 3.4.33 — Описание регистра ALT_FUNCTION_CTRL_G

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
от N · 2 до N · 2+1	BIT_FUNC	Для бита N (N=0 ... 15, номер вывода микросхемы) значение поля определяет используемую альтернативную функцию: 0 — функция 1; 2 — функция 3; 1 — функция 2; 3 — не используется.

Таблица 3.4.34 — Описание регистра ALT_FUNCTION_CTRL_H

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
от N · 2 до N · 2+1	BIT_FUNC	Для бита N (N=0 ... 15, номер вывода микросхемы) значение поля определяет используемую альтернативную функцию: 0 — функция 1; 2 — функция 3; 1 — функция 2; 3 — не используется.

Таблица 3.4.35 — Описание регистра ALT_FUNCTION_CTRL_I

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
от N · 2 до N · 2+1	BIT_FUNC	Для бита N (N=0 ... 15, номер вывода микросхемы) значение поля определяет используемую альтернативную функцию: 0 — функция 1; 2 — функция 3; 1 — функция 2; 3 — не используется.

3.4.1.24 Регистр ALIAS_CTRL

Таблица 3.4.36 — Описание регистра ALIAS_CTRL

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 4	–	
3 – 2	EXTMEM_ALIAS (RW)	алиасинг для внешней памяти имеет значение только если INTMEM_ALIAS=0: 0 — алиасинг внешней памяти по Chip Select 1 (0x0800_0000 — 0x08FF_FFFF); 1 — алиасинг внешней памяти по Chip Select 2 (0x0900_0000 — 0x09FF_FFFF); 2 — алиасинг внешней памяти по Chip Select 3 (0x0A00_0000 — 0x0AFF_FFFF); 3 — алиасинг внешней памяти по Chip Select 4 (0x0B00_0000 — 0x0BFF_FFFF).
1 – 0	INTMEM_ALIAS (RW)	алиасинг для внутренней памяти: 0 — нет алиасинга; 1 — алиасинг только блока 1 (0x2000_0000 — 0x2000_FFFF); 2 — алиасинг только блока 2 (0x2001_0000 — 0x2001_FFFF); 3 — алиасинг обоих блоков (0x2000_0000 — 0x2001_FFFF).

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	
Изм	Лист	№ докум.	Подп.	Дата	
Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.					Лист 42

3.4.1.25 Регистр EDAC_REACTION_CTRL

Таблица 3.4.37 — Описание регистра EDAC_REACTION_CTRL

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 21	–	
20 – 18	CACHE_CRC_ERR	Реакции на соответствующие события: 0 — нет реакции; 1 — прерывание; 2 — исключение; 3 — сброс всего, исключая дебаг; 4 — сброс всего, включая дебаг; 6 – 7 — нет реакции.
17 – 15	INTMEM_SCR_FERR	
14 – 12	INTMEM_SCR_CERR	
11 – 9	INTMEM_FERR	
8 – 6	INTMEM_CERR	
5 – 3	EXTMEM_FERR	
2 – 0	EXTMEM_CERR	

3.4.1.26 Регистр CACHE_MAIN

Таблица 3.4.38 — Описание регистра CACHE_MAIN

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 4	–	
1	CACHE_READY	1 — готов, включен; 0 — нет.
0	CACHE_ENABLE	1 — запрос на включение, инициализацию; 0 — выключен.

3.4.1.27 Регистр SCRUBBER_FERR_ADDR

Таблица 3.4.39 — Описание регистра SCRUBBER_FERR_ADDR

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 0	ADDR	Адрес в адресном пространстве внутренней памяти, по которому блоками скрабберов было обнаружена последняя ошибка памяти типа FERR.

3.4.1.28 Регистр COMMON_FERR_ADDR

Таблица 3.4.40 — Описание регистра COMMON_FERR_ADDR

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 0	ADDR	Адрес в адресном пространстве внутренней либо внешней памяти, по которому блоком исправления ошибок была обнаружена последняя ошибка памяти типа FERR.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.					Лист
					Изм	Лист	№ докум.	Подп.	Дата	
										43

3.4.1.29 Регистр GLOBAL_RESET

Таблица 3.4.41 — Описание регистра GLOBAL_RESET

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 1	–	
0	RESET	Запись «1» инициирует сброс всех блоков микросхемы, включая отладчик.

3.5 Контроллер внутренней памяти

Микросхема содержит внутреннюю память объёмом 128 кБ, реализованную в виде двух блоков размером по 64 кБ. Доступ к внутренней памяти осуществляется с помощью контроллера памяти, расположенного на шине АНВ. Контроллер включает в себя встроенный блок обнаружения и исправления ошибок Error Detection and Correction (EDAC), который исправляет одну битовую ошибку и детектирует наличие двух битовых ошибок на 8 бит данных. Работа контроллера может осуществляться в двух режимах:

- без использования блока исправления ошибок;
- с включенным блоком исправления ошибок.

Включение и выключение блока исправления ошибок осуществляется в регистре общего назначения EDAC_CTRL. Можно отдельно включить или отключить исправление ошибок для каждого из двух блоков внутренней памяти.

В случае возникновения одной битовой ошибки в байте данных контроллер производит автоматическое исправление ошибки при чтении. При этом инкрементируется счётчик ошибок INTMEM_CERR_CNT. В случае возникновения неисправимой ошибки инкрементируется счётчик INTMEM_FERR_CNT в области регистров общего назначения.

Доступ к памяти осуществляется за один такт системной частоты процессора на запись и чтение как с включенным блоком исправления ошибок, так и без его использования.

Также имеется возможность прямого доступа к данным каждого блока памяти и к битам чётности, используемым блоком исправления ошибок. Доступ возможен

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>					Лист
Изм.	Лист	№ докум.	Подп.	Дата						44

путём чтения и записи в специальные области памяти адресного пространства (п. 3.2).

В области внутренней памяти используется аппаратная система исправления и обнаружения ошибок, реализованная с помощью скрабберов. Настройки скрабберов осуществляются с помощью регистров в области регистров общего назначения. Для каждого блока памяти используется свой скраббер. Программно настраивается верхняя граница диапазона адресов, находящихся в области сканирования скраббера (регистры INTMEM_SCR_RNG_ADDR), периодичность сканирования (регистры INTMEM_SCR_PRD_SCAN) и период прерывания скраббером обращений контроллера внутренней памяти (регистры INTMEM_SCR_PRD_STOP). Каждый из двух скрабберов может быть отдельно включен или выключен (регистр INTMEM_SCR_MAIN). Каждый скраббер с указанным периодом производит чтение данных из заданного диапазона и производит исправление данных в случае, если была обнаружена единичная битовая ошибка, либо сигнализирует о наличии неисправимой ошибки в случае повреждения более чем двух бит данных. Ошибки, обнаруженные скрабберами, подсчитываются в отдельных регистрах (регистры EDAC_INTMEM_SCR_CERR, EDAC_INTMEM_SCR_FERR).

Реакция на обнаружение неисправимых ошибок и исправление единичных ошибок также настраивается программно в регистре EDAC_REACTION_CONTROL, в котором отдельно настраивается реакция для блока EDAC и для скрабберов. В зависимости от настроек, в случае возникновения ошибки того или иного типа, может возникать прерывание, высокоприоритетное исключение (немаскируемое прерывание NMI), либо происходить асинхронный сброс микросхемы (включающий, либо не включающий в себя сброс отладчика).

3.6 Контроллер внешней памяти

Контроллер внешней памяти позволяет работать с четырьмя устройствами по интерфейсу асинхронной памяти, т.е. имеет четыре Chip Select. Это могут быть микросхемы памяти SRAM или FLASH, либо другие устройства имеющие такой интерфейс.

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>					Лист
Изм.	Лист	№ докум.	Подп.	Дата						45

Контроллер имеет шину адреса 24 бита, что позволяет адресовать 16 МБ по каждому Chip Select.

Доступ к контроллеру в общем адресном пространстве осуществляется по адресам:

- 0x08000000 – 0x08FFFFFF Chip Select 1;
- 0x09000000 – 0x09FFFFFF Chip Select 2;
- 0x0A000000 – 0x0AFFFFFF Chip Select 3;
- 0x0B000000 – 0x0BFFFFFF Chip Select 4.

Контроллер позволяет выбрать для каждого Chip Select:

- разрядность данных памяти 8 или 16 бит;
- включить или выключить режим исправления ошибок. Режим исправления ошибок требует дополнительно увеличить шину данных внешней памяти до 13 бит для 8-битного режима и 26 бит для 16-битного режима. Режим исправления ошибок позволяет исправить одну ошибку в одном байте данных или обнаружить двойную ошибку в одном байте данных. Настройка блока исправления ошибок производится в регистрах общего назначения EXTMEM_CTRL (Chip Select 1 – 4), количество обнаруженных ошибок подсчитывается в регистрах EXTMEM_CERR_CNT и EXTMEM_FERR_CNT (общие счётчики для всех Chip Select);

- длительность операции чтения от 1 до 8 тактов;

- длительность операции записи от 1 до 8 тактов. Если выбран режим записи 1 такт, то сигнал WEN устанавливается в 0 на полтакта, адрес, сигналы управления и данные выставляются за полтакта минус 1 нс до сигнала WEN, а снимаются через 1 нс после снятия сигнала WEN. Для режимов записи от 2 до 8 тактов, сигнал WEN устанавливается соответственно от 1 до 7 тактов, адрес, сигналы управления и данные выставляются за полтакта до сигнала WEN, и снимаются через полтакта после снятия WEN;

- задержка Chip Select от 0 до 7 тактов. Позволяет выставить задержку сигнала Chip Select (CEN) на определённое число тактов относительно сигналов адреса, данных и прочих сигналов управления, что увеличивает транзакцию на

Изм.	Лист	№ докум.	Подп.	Дата																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																										
------	------	----------	-------	------	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

соответствующее количество тактов. Применяется для подключения нестандартных устройств, например на ПЛИС.

Временные диаграммы записи данных при различных настройках длительности приведены на рисунках 3.6.1, 3.6.2, 3.6.3.

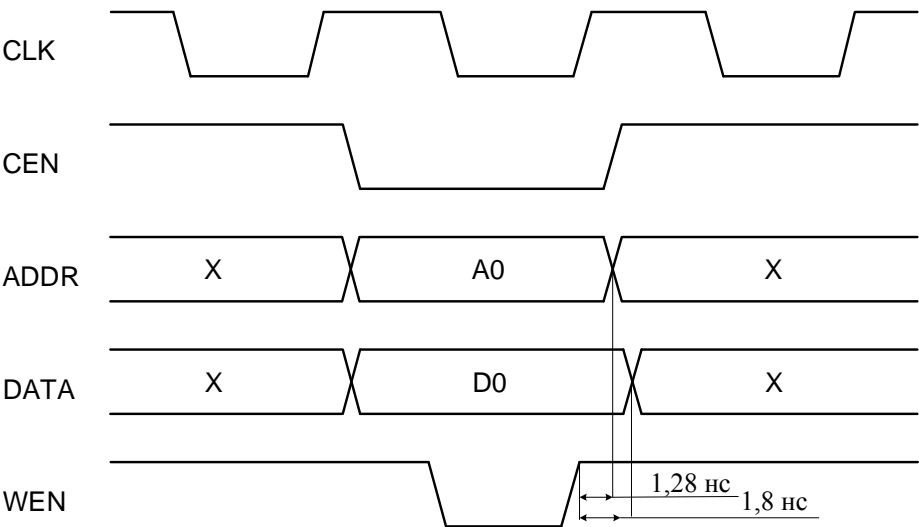


Рисунок 3.6.1 — Транзакция записи при длительности операции записи в 1 такт и задержки Chip Select 0 тактов. Примечание — WE переключается в ноль по заднему фронту, в единицу — по переднему фронту сигнала CLK

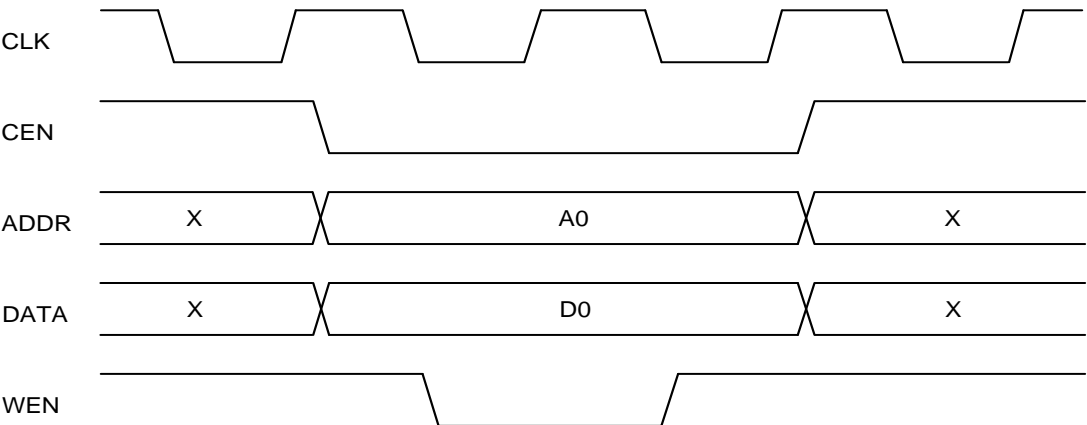


Рисунок 3.6.2 — Транзакция записи при длительности операции записи в 2 такта и задержки Chip Select 0 тактов. Примечание — WE переключается по заднему фронту сигнала CLK

Име. № подл.	Подп. и дата.
Взам. инв. №	Инв. № дубл.
Подп. и дата	
Име. № подл.	

Контроллер имеет следующие особенности:

- программируемая генерация прерываний;
- доступ по маске;
- регистры для перевода вывода в режим альтернативной функции;
- входной сигнал пересинхронизируется через два триггера во избежание метастабильности.

3.7.1 Генерация прерываний

Блок контроллера GPIO поддерживает программную генерацию прерывания по событиям ввода-вывода. Для настройки прерываний используются три регистра с отдельным доступом по установке и снятию значения. Таблица 3.7.1 демонстрирует события, по которым возникают прерывания в зависимости от значений регистров.

Таблица 3.7.1 — Прерывания контроллера GPIO

Interrupt enable[n]	Interrupt polarity[n]	Interrupt type[n]	Событие
0	—	—	Запрет прерывания
1	0	0	Срабатывание по уровню — низкий уровень
1	0	1	Срабатывание по фронту — отрицательный фронт
1	1	0	Срабатывание по уровню — высокий уровень
1	1	1	Срабатывание по фронту — положительный фронт

При возникновении соответствующего события на входе выставляется соответствующий бит регистра INTSTATUS, а также выставляется прерывание в контроллере прерываний Nested Vectory Interrupt Controller (NVIC). Пользователь может сбросить прерывание записью 1 в соответствующий бит регистра INTCLEAR.

3.7.2 Доступ по маске

Доступ к портам ввода-вывода по маске позволяет записывать единичные биты порта за одну операцию записи. При использовании маскированного доступа операция записи в 16-битный порт разбивается на две части:

- запись младших восьми бит порта;
- запись старших бит.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.					Лист 49
					Изм	Лист	№ докум.	Подп.	Дата	

Для доступа по маске к младшим битам используется массив из 256 регистров MASKLOWBYTE, для доступа к старшим — MASKHIGHBYTE.

Например, если требуется одновременно выставить единицы на выходы [1:0] и нули на выходы [7:6], необходимо выполнить доступ по маске к младшим битам. Маска для записи равна соответственно 0xC3 и в соответствующий регистр MASKLOWBYTE должно быть записано значение изменяемых бит: MASKLOWBYTE[0xC3] = 0x03, как показано на рисунке 3.7.1.

Также производится запись старших восьми бит порта ввода вывода. Для записи используется адресное пространство регистров MASKHIGHBYTE (см. рисунок 3.7.2).

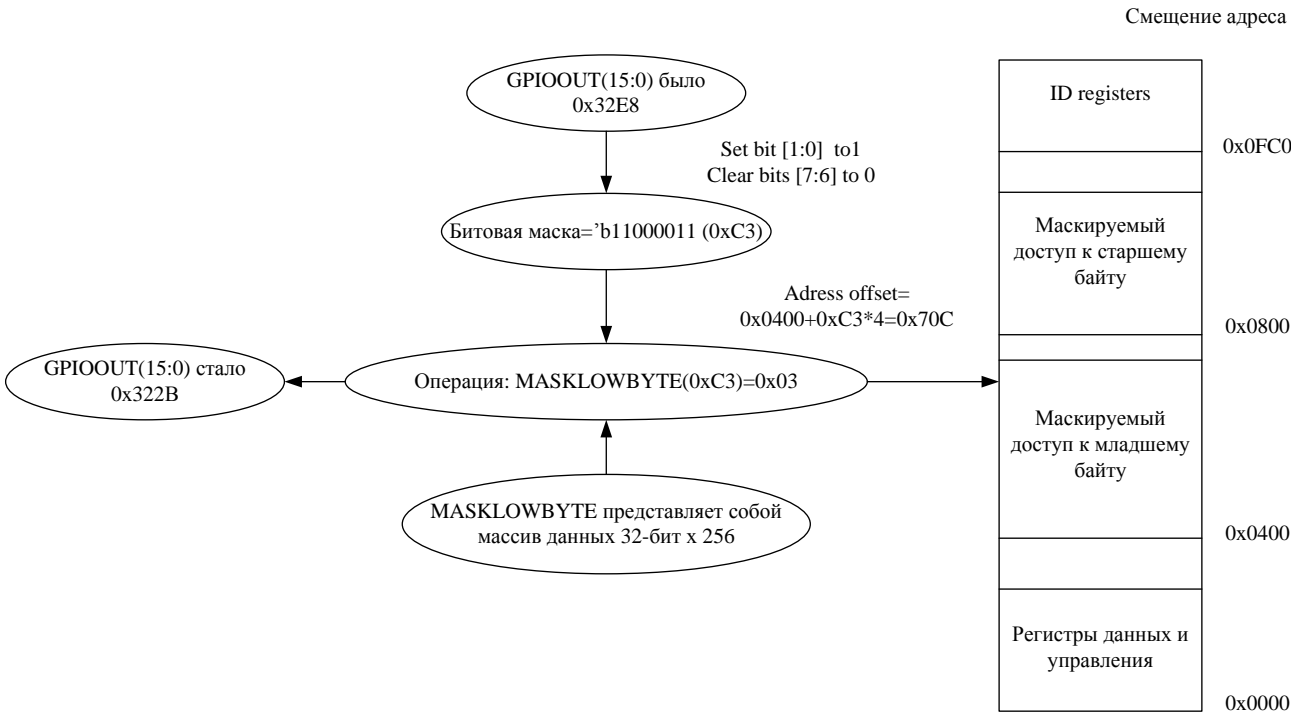


Рисунок 3.7.1 — Операция записи младших бит порта ввода-вывода

Име. № подл.	Подп. и дата.
Взам. инв. №	Инв. № дубл.
Подп. и дата	
Изм	Лист
№ докум.	Подп.
Дата	

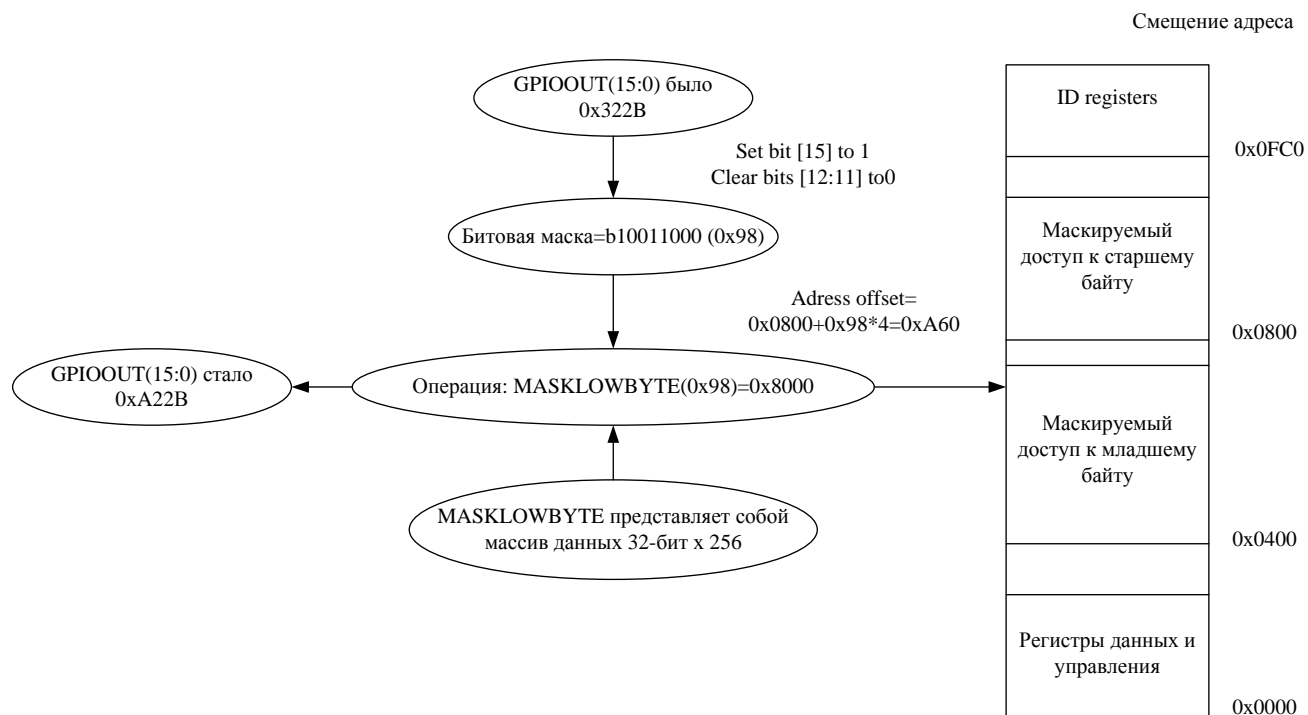


Рисунок 3.7.2 — Операция записи старших бит порта ввода-вывода

3.7.3 Регистры контроллера GPIO

3.7.3.1 Адресное пространство

Назначение регистров контроллера GPIO приведено в таблице 3.7.2.

Таблица 3.7.2 — Регистры контроллера GPIO

Сме- щение	Название регистра	Тип	Значение по сбросу	Описание
1	2	3	4	5
0x0000	DATA	RW	0x----	Регистр данных порта [15:0] По чтению: Данные на входе порта По записи: Значение выхода порта. Чтение записанного значения возможно после задержки в 2 такта системной частоты из-за пересинхронизации входного значения
0x0004	DATAOUT	RW	0x0000	Значение выходного регистра данных [15:0] По чтению: Текущее значение регистра данных По записи: Запись значения регистра данных
0x0008 –0x000c	Резерв	–	–	

Продолжение таблицы 3.7.2

1	2	3	4	5
0x0010	OUTENSET	RW	0x0000	Разрешение выхода порта [15:0] По записи: 1 — Разрешение выхода порта 0 — Выход в третьем состоянии По чтению: 0 — Вывод порта в состоянии вход 1 — Вывод порта в состоянии выход
0x0014	OUTENCLR	RW	0x0000	Сброс ранее установленного значения разрешения порта [15:0]. По записи: 1 — Перевод порта в третье состояние 0 — Не изменяется ранее установленное значение в регистре По чтению: 0 — Вывод порта в состоянии вход 1 — Вывод порта в состоянии выход
0x0018	ALTFUNCSET	RW	Порт A: 0xFFFF Порт B: 0xFFFF Порт C: 0xFFFF Порт D: 0x007F Порт E: 0x0008 Порт F: 0x0000 Порт G: 0x0000 Порт H: 0x00E0 Порт I: 0x0000	По записи: 1 — Перевод вывода в режим альтернативной функции 0 — Не изменяется ранее установленное значение в регистре По чтению: 0 — Вывод в режиме GPIO 1 — Вывод в режиме альтернативной функции
0x001c	ALTFUNCCLR	RW	Порт A: 0xFFFF Порт B: 0xFFFF Порт C: 0xFFFF Порт D: 0x007F Порт E: 0x0008 Порт F: 0x0000 Порт G: 0x0000 Порт H: 0x00E0 Порт I: 0x0000	По записи: 1 — Перевод вывода в режим GPIO 0 — Не изменяется ранее установленное значение в регистре По чтению: 0 — Вывод в режиме GPIO 1 — Вывод в режиме альтернативной функции
0x0020	INTENSET	RW	0x0000	Регистр разрешения прерываний [15:0]: По записи: 1 — Разрешение прерываний 0 — Не изменяется ранее установленное значение в регистре По чтению: 0 — Прерывания запрещены 1 — Прерывания разрешены
0x0024	INTENCLR	RW	0x0000	Сброс регистра разрешения прерываний [15:0]: По записи: 1 — Запрещение прерываний 0 — Не изменяется ранее установленное значение в регистре По чтению: 0 — Прерывания запрещены 1 — Прерывания разрешены

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.

Изм	Лист	№ докум.	Подп.	Дата

Микросхема интегральная 5023BC016

Техническое описание

Версия 2.04 от 01.09.2017 г.

Лист

52

Копировал

Формат А4

Продолжение таблицы 3.7.2

1	2	3	4	5
0x0028	INTTYPESET	RW	0x0000	Регистр типа прерываний [15:0] По записи: 1 — Установка типа прерываний 0 — Не изменяется ранее установленное значение в регистре По чтению: 0 — Прерывания по уровню 1 — Прерывания по фронту
0x002c	INTTYPECLR	RW	0x0000	Сброс регистра типа прерываний [15:0] По записи: 1 — Сброс регистра типа прерываний, 0 — Не изменяется ранее установленное значение в регистре По чтению: 0 — Прерывания по уровню 1 — Прерывания по фронту
0x0030	INTPOLSET	RW	0x0000	Регистр полярности события прерывания [15:0] По записи: 1 — Установка полярности события прерываний 0 — Не изменяется ранее установленное значение в регистре По чтению: 0 — Прерывания по низкому уровню или отрицательному фронту 1 — Прерывания по высокому уровню или положительному фронту
0x0034	INTPOLCLR	RW	0x0000	Сброс регистра полярности события прерывания [15:0] По записи: 1 — Сброс регистра полярности события прерываний 0 — Не изменяется ранее установленное значение в регистре По чтению: 0 — прерывания по низкому уровню или отрицательному фронту 1 — Прерывания по высокому уровню или положительному фронту
0x0038	INTSTATUS, INTCLEAR	RW	0x0000	Регистр сброса прерывания [15:0] По записи: 1 — Сброс соответствующего прерывания 0 — Не изменяется ранее установленное значение в регистре По чтению: [15:0] — Статус прерывания
0x400– 0x7FC	MASKLOWBYTE	RW	0x----	Доступ по маске для младших восьми бит порта. Биты [9:2] адреса регистра используют как битовая маска. [15:8] — не используется. [7:0] — совместно с битами адреса [9:2] используются как маска для битов данных порта ввода-вывода
0x800– 0xBFC	MASKHIGHBYTE	RW	0x----	Доступ по маске для старших восьми бит порта. Биты [9:2] адреса регистра используют как битовая маска. [15:8] — совместно с битами адреса [9:2] используются как маска для битов данных порта ввода-вывода. [7:0] — не используется.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.

Изм	Лист	№ докум.	Подп.	Дата

Микросхема интегральная 5023BC016

Техническое описание
Версия 2.04 от 01.09.2017 г.

Лист

53

Копировал

Формат А4

— оконечное устройство;

— режим монитора (М).

Контроллер последовательного интерфейса по ГОСТ Р 52070 состоит из следующих основных блоков:

— блок АМБА АНВ Master (реализует прямой доступ к памяти (DMA));

— блок контроллера шины (реализует функции контроллера в режиме контроллера шины);

— блок оконечного устройства (реализует функции контроллера в режиме оконечного устройства);

— блок монитора шины (реализует функции контроллера в режиме Монитора (М);

— передатчик каналов А и В (энкодер магистрального последовательного интерфейса с центральным управлением);

— приёмники А, В, С и D (декодер магистрального последовательного интерфейса с центральным управлением); используются 2 декодера на один канал: на канал А — декодеры А и В, а на канал В — С и D; декодер А принимает сигнал как А+ положительная составляющая дифференциального сигнала и А- как отрицательная составляющая дифференциального сигнала; а декодер В наоборот, А+ как отрицательная составляющая дифференциального сигнала и А — как положительная составляющая дифференциального сигнала; данная реализация позволяет принимать сигналы с любой полярностью, и если сигналы физически были перепутаны, то это не повлияет на работу устройства;

— регистры управления контроллером (Registers).

3.8.2 Регистры контроллера

Распределение регистров контроллера последовательного интерфейса по ГОСТ Р 52070 приведено в таблице 3.8.1.

Назначения разрядов регистров контроллера шин приведены в таблицах 3.8.2 – 3.8.11.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	<div>Микросхема интегральная 5023BC016</div> <div>Техническое описание</div> <div>Версия 2.04 от 01.09.2017 г.</div>					Лист
Изм.	Лист	№ докум.	Подп.	Дата						55

Назначения разрядов регистров оконечных устройств приведены в таблицах 3.8.12 – 3.8.15.

Назначения разрядов регистров монитора приведены в таблицах 3.8.16 – 3.8.19.

Назначения разрядов регистров монитора приведены в таблицах 3.8.20 – 3.8.23.

Таблица 3.8.1 — Распределение регистров контроллера последовательного интерфейса по ГОСТ Р 52070

Смещение	Название регистра	Тип	Значение по сбросу	Описание
0x00	BCONE0	W/R	0x00000000	Первый регистр группы 1 КШ
0x04	BCONE1	W/R	0x00000000	Второй регистр группы 1 КШ
0x08	BCONE2	W/R	0x00000000	Третий регистр группы 1 КШ
0x0C	BCONE3	RO	0x00000000	Четвертый регистр группы 1 КШ
0x10	BCONE4	RO	0x00000000	Пятый регистр группы 1 КШ
0x14	BCTWO0	W/R	0x00000000	Первый регистр группы 2 КШ
0x18	BCTWO1	W/R	0x00000000	Второй регистр группы 2 КШ
0x1C	BCTWO2	W/R	0x00000000	Третий регистр группы 2 КШ
0x20	BCTWO3	RO	0x00000000	Четвертый регистр группы 2 КШ
0x24	BCTWO4	RO	0x00000000	Пятый регистр группы 2 КШ
0x28	EP0	W/R	0x00000000	Регистр настроек ОУ 0
0x2C	EP1	W/R	0x00000008	Регистр настроек ОУ 1
0x30	EP2	RO	0x00000000	Регистр настроек ОУ 2
0x34	EP3	RO	0x00000000	Регистр настроек ОУ 3
0x38	SETTING0	W/R	0x00000000	Регистр общих настроек 0
0x3C	SETTING1	W/R	0x00000000	Регистр общих настроек 1
0x40	SETTING2	W/R	0x00000000	Регистр общих настроек 2
0x44	SETTING3	W/R	0x00000000	Регистр общих настроек 3
0x48	MONITOR0	W/R	0x00000000	Регистр настроек монитора 0
0x4C	MONITOR1	W/R	0x00000000	Регистр настроек монитора 1
0x50	MONITOR2	W/R	0x00000000	Регистр настроек монитора 2
0x54	MONITOR3	RO	0x00000000	Регистр настроек монитора 3
0x58	INTERRUPT	W/R	0x00000000	Регистр прерываний

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Изм	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 56
-----	------	----------	-------	------	---	------------

3.8.3.1 Регистр BCONE0

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 26	–	
25 – 21	EP_ADDR0	адрес ОУ, которому предназначается командное слово (КС). Если адрес ОУ равен 111112, то КС является групповым
20 – 16	SUB_ADDR0	подадрес ОУ, которому предназначается КС. Значения подадреса равные 111112 или 000002 являются признаком режима управления и не должны использоваться
15	–	
14 – 10	NUM	количество передаваемых или принимаемых слов данных(СД). Также в режиме управления содержит код команды
9 – 5	EP_ADDR1	используется только для транзакций форматов 3 и 8 — адрес ОУ для второго КС
4 – 0	SUB_ADDR1	используется только для транзакций форматов 3 и 8 — подадрес ОУ для второго КС

Таблица 3.8.3 — Описание регистра BCONE1

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 0	DMA_ADDR	содержит адрес первого байта области памяти с СД. Младшие два бита регистра всегда равны нулю.

Таблица 3.8.4 — Описание регистра BCONE2

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 16	COM_DATA	только для транзакций форматов 6 и 10 — содержит СД, которое необходимо передать для этих транзакций
15 – 9	–	
8	ENA	запуск начала транзакции; при записи 1 блок начинает транзакцию; при чтении 1 после записи 1, блок не начал данной транзакции
7 – 5	–	
4	MANAGE_CODE	задает формат признака режима управления в командном слове. При значении MANAGE_CODE = 0, признак режима управления будет иметь значение 000002, иначе — 111112
3 – 0	TYPE_TRANS	код транзакции

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.

Инв. № подл.	Подп. и дата

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 16	REPLY	Ответное слово (OC) на КС транзакции
15 – 0	REPLY_WORD	только для транзакции формата 5; содержит СД, которое передается с OC

Инв. № подл.	
--------------	--

Инв. № подл.	Подп. и дата

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 17	–	
16	TIMEOUT	бит тайм-аута; чтение 1 означает, что в транзакции произошёл таймаут (не пришло ОС или СД от ОУ, необходимо смотреть ОС)
15 – 9	–	
8	ERROR	бит ошибки; чтение 1 означает, что в транзакции произошла ошибка (несоответствие бита четности принимаемых данных)
7 – 1	–	
0	STATUS	бит статуса; чтение 1 означает, что транзакция в процессе выполнения

Инв. № подл.	Подп. и дата

Инв. № подл.	
--------------	--

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 26	–	
25 – 21	EP_ADDR0	адрес ОУ, которому предназначается командное слово; если адрес ОУ равен 111112, то поле является групповым КС
20 – 16	SUB_ADDR0	подадрес ОУ, которому предназначается КС. Значения подадреса равные 111112 или 000002 являются признаком режима управления и не должны использоваться.
15	–	
14 – 10	NUM	количество передаваемых или принимаемых слов данных; в режиме управления поле также содержит код команды
9 – 5	EP_ADDR1	адрес ОУ для второго КС — используется только для транзакций форматов 3 и 8
4 – 0	SUB_ADDR1	подадрес ОУ для второго КС используется только для транзакций форматов 3 и 8

3.8.3.7 Регистр BCTWO1

Таблица 3.8.8 — Описание регистра BCTWO1

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 0	DMA_ADDR	содержит адрес первого байта области памяти с СД. Младшие два бита регистра всегда равны нулю

3.8.3.8 Регистр BCTWO2

Таблица 3.8.9 — Описание регистра BCTWO2

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 16	COM_DATA	СД, которое нужно передать для этих транзакций; используется только для транзакций форматов 6 и 10
15 – 19	–	
8	ENA	запуск начала транзакции; при записи 1 блок начинает транзакцию; при чтении 1 после записи 1 блок не начал данной транзакции
7 – 5	–	
4	MANAGE_CODE	задает формат признака режима управления в командном слове. При значении MANAGE_CODE = 0, признак режима управления будет иметь значение 000002, иначе — 11112
3 – 0	TYPE_TRANS	код транзакции

3.8.3.9 Регистр BCTWO3

Таблица 3.8.10 — Описание регистра BCTWO3

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 16	REPLY	ОС на КС транзакции
15 – 0	REPLY_WORD	только для транзакции формата 5; Содержит СД, которое передается с ОС

3.8.3.10 Регистр BCTWO4

Таблица 3.8.11 — Описание регистра BCTWO4

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 17	–	
16	TIMEOUT	бит тайм-аута; чтение 1 означает, что в транзакции произошел таймаут (не пришло ОС или СД от ОУ, необходимо использовать ОС)
15 – 9	–	
8	ERROR	бит ошибки; чтение 1 означает, что в транзакции произошла ошибка (несоответствие бита четности принимаемых данных)
7 – 1	–	
0	STATUS	бит статуса; чтение 1 означает, что транзакция в процессе выполнения.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.					Лист
					Изм	Лист	№ докум.	Подп.	Дата	
										59

3.8.4 Регистры оконечного устройства

3.8.4.1 Регистр EP0

Таблица 3.8.12 — Описание регистра EP0

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 0	DMA_EP	общая память для всех подадресов. Содержит адрес первого байта области памяти ОУ. Чтобы получить доступ к определенному подадресу, нужно сместить указатель памяти на $64 \cdot k$, где k — подадрес. Младшие два бита регистра всегда равны нулю

3.8.4.2 Регистр EP1

Таблица 3.8.13 — Описание регистра EP1

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 16	RDATA	содержит СД, которое необходимо передать для этих транзакций (используется только для транзакции формата 5)
15 – 10	—	
9	ERROR	признак «Ошибка в сообщении»
8	SREQ	признак «Запрос на обслуживание» в ОС ОУ
7 – 5	RESERV	резерв в ОС ОУ
4	RXGROUP	признак «Принята групповая команда»
3	BUSY	признак «Абонент занят» в ОС ОУ, при сбросе равен «1»
2	FEP0	признак «Неисправность абонента» в ОС ОУ
1	MAN	признак «Принято управление интерфейсом» в ОС ОУ
0	FEP1	признак «Неисправность ОУ» в ОС ОУ

3.8.4.3 Регистр EP2

Таблица 3.8.14 — Описание регистра EP2

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 2	—	
1	EP_COM_TIMEOUT	бит тайм-аута; чтение «1» означает, что в транзакции произошел таймаут (в транзакции недостаточно СД)
0	EP_COM_ERR	чтение «1» означает, что в транзакции была ошибка

Изм.	Лист	№ докум.	Подп.	Дата	Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	4	RXGROUP	признак «Принята групповая команда»
										3	BUSY	признак «Абонент занят» в ОС ОУ, при сбросе равен «1»
										2	FEP0	признак «Неисправность абонента» в ОС ОУ
										1	MAN	признак «Принято управление интерфейсом» в ОС ОУ
										0	FEP1	признак «Неисправность ОУ» в ОС ОУ

3.8.4.3 Регистр EP2

Таблица 3.8.14 — Описание регистра EP2

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 2	–	
1	EP_COM_TIMEOUT	бит тайм-аута; чтение «1» означает, что в транзакции произошел таймаут (в транзакции недостаточно СД)
0	EP_COM_ERR	чтение «1» означает, что в транзакции была ошибка

Изм.	Лист	№ докум.	Подп.	Дата	Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	Микросхема интегральная 5023BC016		Лист
										Техническое описание		60
											Версия 2.04 от 01.09.2017 г.	

3.8.4.4 Регистр EP3

Таблица 3.8.15 — Описание регистра EP3

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 16	EP_COM	КС принятое ОУ; используется в режиме ОУ
15 – 0	EP_COM_DATA	только для транзакций формата 6 и 10; содержит принятое СД этих транзакций.

3.8.5 Общие регистры настроек

3.8.5.1 Регистр SETTING0

Таблица 3.8.16 — Описание регистра SETTING0

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 24	SYS_CLK	значение это байта вычисляется по формуле SYSTEM_CLOCK — «1», где SYSTEM_CLOCK — это системная частота в МГц; значение должно быть установлено для вычисления задержек
23 – 17	—	
16	CHANNEL	0 — рабочий канал для КШ А 1 — рабочий канал для КШ В
15 – 10	—	
9 – 8	MODE	задает режим работы блока: 00 — блок отключён; 01 — блок функционирует в режиме КШ; 10 — блок функционирует в режиме ОУ; 11 — блок функционирует в режиме М.
7 – 5	—	
4 – 0	EP_ADDR	задает адрес ОУ

3.8.5.2 Регистр SETTING1

Таблица 3.8.17 — Описание регистра SETTING1

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
1	2	3
31 – 28	—	
27 – 24	TIME	задает время паузы между наступлением «тишины» на линии и следующим сообщением в микросекундах; необходимо для работы в режиме КШ и ОУ: в режиме КШ — время перед отправкой следующей транзакции, в режиме ОУ — время перед отправкой ОС; общая пауза вычисляется как сумма TIME + NOACTTIME (см. регистр SETTING2) и должна быть не менее 4 мкс

Име. № подл.	Подп. и дата.
Взам. инв. №	Инв. № дубл.
Подп. и дата	
Име. № подл.	

Изм.	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 61
------	------	----------	-------	------	---	------------

Продолжение таблицы 3.7.2

1	2	3
23 – 16	RWAIT	задает время ожидания следующего слова в режиме работы КШ и ОУ в микросекундах; должно быть не менее 21 мкс; необходимо для определения тайм-аута в сообщениях
15 – 8	EDGE	задает время запрещенного состояния дифференциального сигнала в тактах системной частоты блока; должно быть не менее 0,2 мкс
7 – 0	–	зарезервировано

3.8.5.3 Регистр SETTING2

Таблица 3.8.18 — Описание регистра SETTING2

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 24	BCWAITREPLY	задает время ожидания ОС в режиме работы КШ в микросекундах; должно быть не менее 20 мкс
23 – 16	NOACTTIME	задает время, в течение которого декодер определяет, что на линии нет активности; задается в тактах системной частоты; время выбирается в зависимости от помех на линии и должно быть не менее 1,3 мкс; необходимо для работы декодера
15 – 14	–	
13 – 10	DECDIS	биты отключения декодеров; 10 бит — отключает декодер А; 11 бит — отключает декодер В; 12 бит — отключает декодер С; 13 бит — отключает декодер D.
9	INCWITH	бит выбора появления сигнала разрешения передачи по последовательному интерфейсу. Если бит равен 0, то сигнал разрешения передачи появляется вместе с дифференциальным сигналом. Если бит равен 1, то сигнал разрешения передачи появляется на 0,5 мкс раньше дифференциального сигнала, и заканчивается позже на 0,5 мкс после дифференциального сигнала.
8	INVTX	бит инвертирования дифференциального сигнала при передаче, т.е. D+ в D-, а D- в D+
7 – 0	500NS	количество тактов системной частоты равно 0,5 мкс; необходимо для работы энкодера магистрального последовательного интерфейса с центральным управлением

3.8.5.4 Регистр SETTING3

Таблица 3.8.19 — Описание регистра SETTING3

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
1	2	3
31 – 24	2250NS	количество тактов системной частоты равно 2,25 мкс; необходимо для работы декодера
23 – 16	1750NS	количество тактов системной частоты равно 1,75 мкс; необходимо для работы декодера

Изм.	Лист	№ докум.	Подп.	Дата	15 – 10	DECDIS	11 бит — отключает декодер B; 12 бит — отключает декодер C; 13 бит — отключает декодер D.
					9	INCWITH	бит выбора появления сигнала разрешения передачи по последовательному интерфейсу. Если бит равен 0, то сигнал разрешения передачи появляется вместе с дифференциальным сигналом. Если бит равен 1, то сигнал разрешения передачи появляется на 0,5 мкс раньше дифференциального сигнала, и заканчивается позже на 0,5 мкс после дифференциального сигнала.
					8	INVTX	бит инвертирования дифференциального сигнала при передаче, т.е. D+ в D-, а D- в D+
					7 – 0	500NS	количество тактов системной частоты равно 0,5 мкс; необходимо для работы энкодера магистрального последовательного интерфейса с центральным управлением

3.8.5.4 Регистр SETTING3

Таблица 3.8.19 — Описание регистра SETTING3

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
1	2	3
31 – 24	2250NS	количество тактов системной частоты равно 2,25 мкс; необходимо для работы декодера
23 – 16	1750NS	количество тактов системной частоты равно 1,75 мкс; необходимо для работы декодера

					Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 62
Изм.	Лист	№ докум.	Подп.	Дата		

Продолжение таблицы 3.8.19

1	2	3
15 – 8	1500NS	количество тактов системной частоты равное 1,5 мкс; необходимо для работы декодера
7 – 0	1250NS	количество тактов системной частоты равное 1,25 мкс; необходимо для работы декодера

3.8.6 Регистры монитора

3.8.6.1 Регистр MONITOR0

Таблица 3.8.20 — Описание регистра MONITOR0

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 0	DMA_MONITOR_ADDR	адрес первого байта области памяти, выделенной для монитора. Младшие два бита регистра всегда равны нулю

3.8.6.2 Регистр MONITOR1

Таблица 3.8.21 — Описание регистра MONITOR1

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 0	MEMORY_SIZE	количество 32 битных слов, выделенных для монитора

3.8.6.3 Регистр MONITOR2

Таблица 3.8.22 — Описание регистра MONITOR2

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 0	THRESHOLD	количество записанных слов, при котором срабатывает прерывание монитора

3.8.6.4 Регистр MONITOR3

Таблица 3.8.23 — Описание регистра MONITOR3

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 0	MONCOUNT	количество записанных слов

Име. № подл.	Подп. и дата.	Име. № дубл.	Подп. и дата.	Взам. инв. №	Име. инв. №	Изм	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 63

3.8.7 Регистр INTERRUPT

Описание регистра SETTING2 указаны в таблице 3.8.24.

Таблица 3.8.24 — Описание регистра SETTING2

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 4	–	
3	MON_INT	прерывание монитора; прерывание срабатывает, когда указатель записи достигает адреса в памяти DMA_MONITOR_ADDR + THRESHOLD
2	EP_INT1	прерывание ОУ - прерывание окончания транзакции в ОУ
1	EP_INT0	прерывание ОУ - прерывание по приёму КС
0	BC_INT	прерывание КШ - прерывание окончания транзакции в КШ

3.8.8 Описание функционирования контроллера последовательного интерфейса по ГОСТ Р 52070

3.8.8.1 Общие настройки контроллера последовательного интерфейса по ГОСТ Р 52070

Для работы контроллера последовательного интерфейса по ГОСТ Р 52070 в любом режиме необходимо произвести настройку блока. За настройку отвечают регистры SETTING0, SETTING1, SETTING2 и SETTING3.

В поле SYS_CLK регистра SETTING0 нужно записать значение системной тактовой частоты в МГц. Для работы блока системная тактовая частота должна быть целым числом, кратна 2 и больше или равно 8 МГц.

В поле CHANNEL нужно записать «0», если интерфейс будет работать по каналу А, или «1», если интерфейс будет работать по каналу В.

В поле RWAIT задать время не менее 21 мкс.

В поле EDGE регистра SETTING1 следует задать максимальное время запрещенного состояния. Запрещённое состояние — это состояние дифференциального сигнала, когда обе линии находятся в логическом нуле. На рисунке 3.8.2 время запрещённого состояния показано как tRG. Данное время задается в тактах приёмной частоты.

Ине. № подл.	Подп. и дата.	Ине. № дубл.	Взам. инв. №	Подп. и дата	Ине. № подл.	Изм	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 64
--------------	---------------	--------------	--------------	--------------	--------------	-----	------	----------	-------	------	---	------------

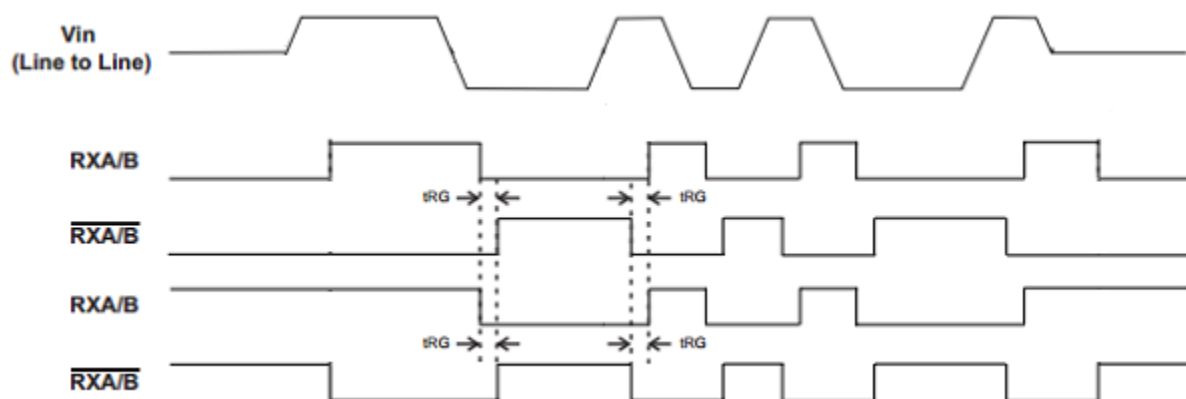


Рисунок 3.8.2 — Временная диаграмма дифференциального сигнала на входе приёмника

В регистре SETTING2 следует задать поле 500NS, т.к. это необходимо для работы энкодера интерфейса. Также в полях NOACTTIME и BCWAITREPLY записать рекомендуемые значения.

Также в регистре SETTING3 необходимо задать времена 1,25; 1,5; 1,75 и 2,25 мкс, которые необходимы для работы декодера.

3.8.8.2 Принцип работы декодера

Чтобы понять работу декодера рассмотрим рисунок 3.8.3, на котором представлен стандартный синхросигнал и несколько битов командного слова без искажений.

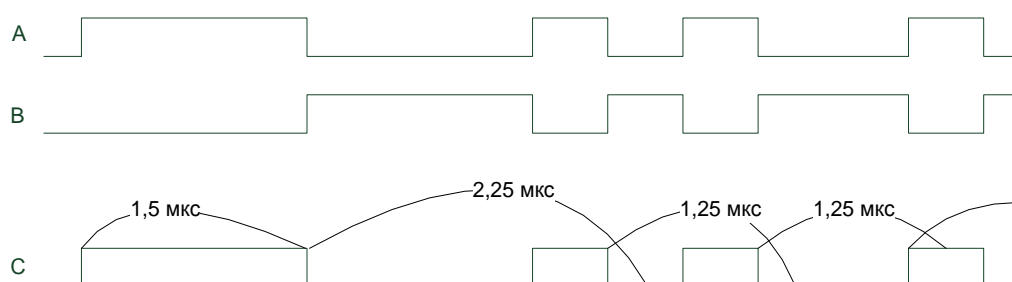


Рисунок 3.8.3 — Результирующий сигнал с первым битом данных равным логической единице.



Рисунок 3.8.4 — Результирующий сигнал с первым битом данных равным логическому нулю

Име. № подл.	Подп. и дата.
Взам. инв. №	Инв. № дубл.
Подп. и дата	
Име. № подл.	

Изм.	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 65
------	------	----------	-------	------	---	------------

На рисунке представлены дифференциальный сигнал, состоящий из A(+) и B(-) сигналов, и его результирующий сигнал C. Результирующий сигнал C формируется по следующему правилу

$$C = \begin{cases} 1, \text{ если } A = "1" \text{ и } B = "0" \\ 0, \text{ если } A = "0" \text{ и } B = "1" \end{cases} \quad (1)$$

Следует сразу указать, что при запрещённом состоянии (A = 1 и B = 1 или A = 0 и B = 0) C не изменяется и сохраняет предыдущее значение.

Декодирование начинается с поиска синхросигнала командного слова — поиск положительной полуволны длительностью 1,5 мкс, и отрицательной полуволны — 1,5 мкс. Но, допуская искажение сигнала при распространении, положительная и отрицательная полуволны синхросигнала должны быть более 1,25 мкс.

При первом бите данных равном 0, отрицательная полуволна синхросигнала сливается с первым битом данных, и образует отрицательную полуволну длительностью в 2 мкс, как показано на рисунке 3.8.4. Поэтому при детектировании синхросигнала захватывается и первый бит данных. Именно из-за этого было выбрано время 2,25 мкс.

Сам процесс детектирования битов данных происходит посредством подсчёта суммы фронтов сигнала (не имеет значение какие фронты — положительные или отрицательные) на промежутке времени. Первый бит данных детектируется на промежутке 2,25 мкс, последующие на 1,25 мкс.

Если на первом бите данных фронт был 1 раз, то это был логический ноль, если 2 раза — логическая единица. Если 0 раз или больше 2-х, то это ошибка детектирования, и процесс детектирования сбрасывается.

На последующих битах данных детектирование зависит от предыдущего бита. Если предыдущий бит равен логическому нулю и по истечении времени был обнаружен только 1 фронт, то текущий бит равен логической единице; если было 2 фронта, то текущий бит равен логическому нулю. Если предыдущий бит равен логической единице и по истечении времени был обнаружен только 1 фронт, то

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>					Лист
Изм.	Лист	№ докум.	Подп.	Дата						66

текущий бит равен логическому нулю; если было 2 фронта, то текущий бит равен логической единице.

В случае искажений при распространении форма результирующего сигнала меняется. На рисунке 3.8.5 представлены сигналы с искажениями и их результирующий сигнал.

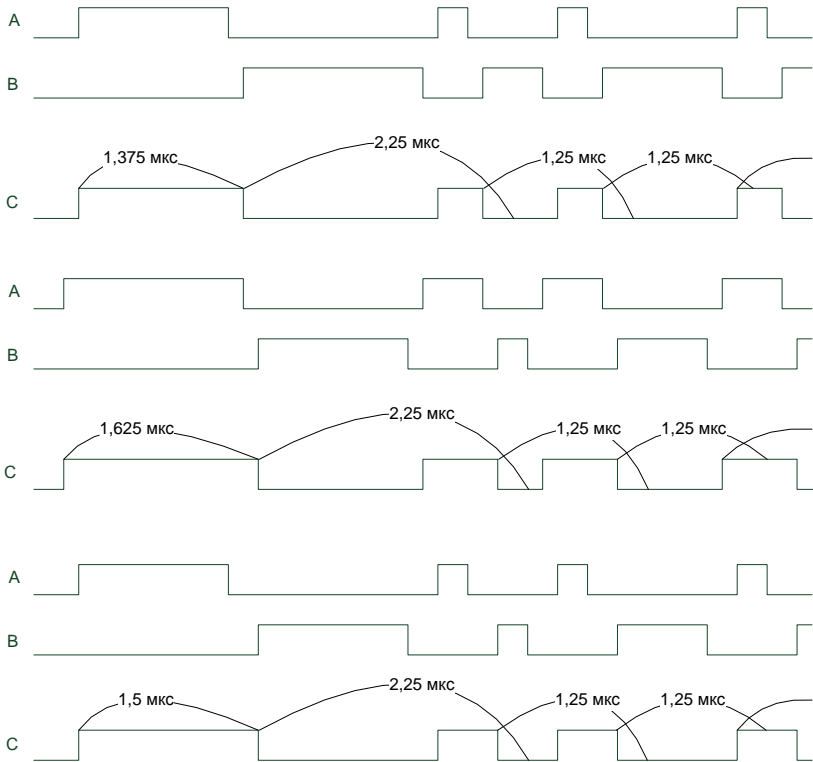


Рисунок 3.8.5 — Возможные искажения сигналов при приёме и их результирующая

На рисунке Рисунок 3.8.5 представлены 3 случая искажений:

- сигнал на линии А имеет более пологий фронт, и после приёмопередатчика это видно, как сужение сигнала, в следствие чего появляется запрещённое состояние;
- сигнал на линии В имеет более пологий фронт;
- сигналы А и В имеют пологие фронты.

Как видно из рисунка 3.8.5, выбранное время 2,25 и 1,25 мкс позволяет принимать в широком диапазоне искажений.

Далее на рисунке 3.8.6 представлены диаграммы сигналов по завершении приёма командного слова. Какое следующее слово: командное или слово данных,

Подп. и дата.	
Инв. № дубл.	
Взам. инв. №	
Подп. и дата	
Инв. № подл.	

					Микросхема интегральная 5023BC016	Лист
					Техническое описание	
					Версия 2.04 от 01.09.2017 г.	67
Изм	Лист	№ докум.	Подп.	Дата		

определяется относительно последнего принятого бита. Если последний бит равен логической единице, то ожидается положительный фронт, и если положительный фронт пришёл по истечении 1,5 мкс, то это слово данных, иначе это следующее командное слово. Если последний бит равен логическому нулю, то ожидается отрицательный фронт, и если отрицательный фронт пришёл по истечении 1,5 мкс, то это следующее командное слово, иначе это слово данных.

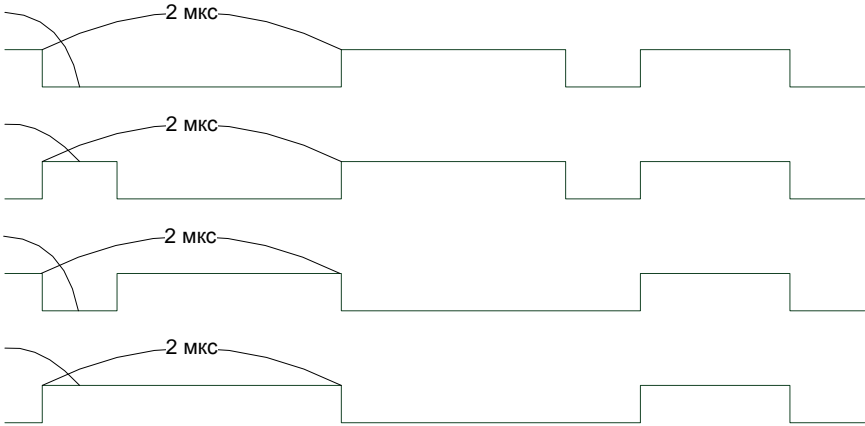


Рисунок 3.8.6 — Диаграммы сигналов по завершении приёма командного слова
Декодирование следующих слов происходит также, как и первое.

Следует подчеркнуть особенность работы декодера: декодер принимает данные не по словам, а по транзакциям, т.е. при любой ошибке на линии декодер останавливает декодирование, выставляет ошибку и ждет «тишины» на линии, после этого он начинает декодировать снова.

3.8.8.3 Функционирование контроллера последовательного интерфейса по ГОСТ Р 52070 в режиме КШ

Для работы блока в режиме КШ, необходимо:

- задать время паузы между сообщениями (см. регистр SETTING1);
- задать время ожидания следующего слова (см. регистр SETTING1);
- задать время ожидания ОС от ОУ (см. регистр SETTING2);
- задать времена 1,25; 1,5; 1,75 и 2,25 мкс (см. регистр SETTING3);
- задать времена 0,5 мкс (см. регистр SETTING2);
- задать поле NOACTTIME (см. регистр SETTING2);
- установить режим работы блока как КШ (см. регистр SETTING0).

Для выполнения транзакций в режиме КИШ в блоке существует две идентичные группы регистров (группа 1: BCONE0, BCONE1, BCONE2, BCONE3 и группа 2: BCTWO0, BCTWO1, BCTWO2, BCTWO3).

Для каждой из групп регистров для начала транзакции необходимо произвести ряд действий:

— проверить наличие признака захвата транзакции в блоке (см. поле ENA в регистре BCONE2 или BCTWO2) и статус транзакции (см. поле STATUS в регистре BCONE3 или BCTWO3). Если поле ENA равно «1», то транзакция находится в очереди транзакций;

— если поле ENA равно «0», но поле STATUS равно «1», то транзакция в настоящий момент передается. Если поле ENA равно «0» и поле STATUS равно «0», то в данную группу регистров разрешается записывать данные для новой транзакции;

— определить канал работы блока (А или В), (см. регистр SETTING0);

— задать адрес первого байта области памяти с СД (см. регистр BCONE1);

— задать адрес ОУ, количество слов СД, формат транзакции и т.д.

3.8.8.3.1 Формирование транзакций для каждого формата в отдельности

Формат транзакции задается в TYPE_TRANS регистре BCONE2 или BCTWO2. Для запуска транзакции любого формата необходимо записать 1 в ENA в регистре BCONE2 или BCTWO2. Таблица 3.8.25 содержит описание форматов.

Таблица 3.8.25 — Описание форматов

Номер формата	Описание
1	2
1	Передача данных от КИШ к ОУ, TYPE_TRANS = 0. Установить адрес ОУ (поле EP_ADDR0), подадрес ОУ (поле SUB_ADDR0), количество слов СД (поле NUM) в регистре BCONE0 или BCTWO0
2	Передача данных от ОУ к КИШ, TYPE_TRANS = 1. Установить адрес ОУ (поле EP_ADDR0), подадрес ОУ (поле SUB_ADDR0), количество слов СД (поле NUM) в регистре BCONE0 или BCTWO0
3	Передача данных от ОУ к ОУ, TYPE_TRANS = 2. Установить адрес приёмного ОУ (поле EP_ADDR0), подадрес приёмного ОУ (поле SUB_ADDR0), количество слов СД (поле NUM), адрес передающего ОУ (поле EP_ADDR1), подадрес передающего ОУ (поле SUB_ADDR1) в регистре BCONE0 или BCTWO0

Ине. № подл.	Подп. и дата.	Ине. № дубл.	Взам. инв. №	Подп. и дата.	<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>					Лист
Изм.	Лист	№ докум.	Подп.	Дата						69

Продолжение таблицы 3.8.25

1	2
4	Передача КУ, TYPE_TRANS = 3. Установить адрес ОУ (поле EP_ADDR0), код команды (поле NUM) в регистре BCONE0 или BCTWO0
5	Передача КУ и приём СД от ОУ, TYPE_TRANS = 4. Установить адрес ОУ (поле EP_ADDR0), код команды (поле NUM) в регистре BCONE0 или BCTWO0
6	Передача КУ с СД к ОУ, TYPE_TRANS = 5. Установить адрес ОУ (поле EP_ADDR0), код команды (поле NUM) в регистре BCONE0 или BCTWO0, и задать СД (поле COM_DATA) в регистре BCONE2 или BCTWO2
7	Групповая передача данных от КИШ к ОУ, TYPE_TRANS = 6. Установить подадрес ОУ (поле SUB_ADDR0), количество слов СД (поле NUM) в регистре BCONE0 или BCTWO0
8	Групповая передача данных от ОУ к ОУ, TYPE_TRANS = 7. Установить подадрес приёмного ОУ (поле SUB_ADDR0), количество слов СД (поле NUM), адрес передающего ОУ (поле EP_ADDR1), подадрес передающего ОУ (поле SUB_ADDR1) в регистре BCONE0 или BCTWO0
9	Групповая передача КУ, TYPE_TRANS = 8. Установить код команды (поле NUM) в регистре BCONE0 или BCTWO0
10	Групповая передача КУ с СД к ОУ, TYPE_TRANS = 9. Установить код команды (поле NUM) в регистре BCONE0 или BCTWO0, и задать СД (поле COM_DATA) в регистре BCONE2 или BCTWO2

Для транзакций формата 1 – 6 КИШ принимает ОС от ОУ. ОС хранится в REPLY в регистре BCONE3 или BCTWO3. Также для транзакций этих форматов есть 2 бита состояний: ERROR и TIMEOUT в регистре BCONE4 или BCTWO4. Признак ERROR означает, что в транзакции произошла ошибка чётности принятых данных в КИШ. Признак TIMEOUT означает, что произошел таймаут, КИШ не принял ОС или принял некорректное количество СД.

Для транзакций формата 5 КИШ принимает ОС с СД. СД находится в поле REPLY_WORD в регистре BCONE3 или BCTWO3.

Для транзакций формата 3 КИШ принимает ОС от передающего ОУ.

3.8.8.4 Функционирование контроллера последовательного интерфейса по ГОСТ Р 52070 в режиме ОУ

Для работы блока в режиме ОУ, необходимо:

- задать время паузы между сообщениями (см. регистр SETTING1);
- задать время ожидания следующего слова (см. регистр SETTING1);
- задать время ожидания ОС от ОУ (см. регистр SETTING2);
- задать времена 1,25; 1,5; 1,75 и 2,25 мкс (см. регистр SETTING3);

Изм.	Лист	№ докум.	Подп.	Дата	<div>Микросхема интегральная 5023BC016</div> <div>Техническое описание</div> <div>Версия 2.04 от 01.09.2017 г.</div>		<div>Лист</div> <div>70</div>
Ине. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	<div>о означает, что в транзакции произошла ошибка чётности принятых данных в КШ. Признак TIMEOUT означает, что произошел таймаут, КШ не принял ОС или принял некорректное количество СД.</div> <div>Для транзакций формата 5 КШ принимает ОС с СД. СД находится в поле REPLY_WORD в регистре BCONE3 или BCTWO3.</div> <div>Для транзакций формата 3 КШ принимает ОС от передающего ОУ.</div> <div>3.8.8.4 Функционирование контроллера последовательного интерфейса по ГОСТ Р 52070 в режиме ОУ</div> <div>Для работы блока в режиме ОУ, необходимо:</div> <div><div>— задать время паузы между сообщениями (см. регистр SETTING1);</div><div>— задать время ожидания следующего слова (см. регистр SETTING1);</div><div>— задать время ожидания ОС от ОУ (см. регистр SETTING2);</div><div>— задать времена 1,25; 1,5; 1,75 и 2,25 мкс (см. регистр SETTING3);</div></div>		

- задать времена 0,5 мкс (см. регистр SETTING2);
- задать поле NOACTTIME (см. регистр SETTING2);
- задать адрес первого байта области памяти с СД (см. регистр EP0);
- задать адрес ОУ (см. регистр SETTING0);
- задать настройки для битов ОС (см. регистр EP1);
- установить режим работы блока как ОУ (см. регистр SETTING0).

Принятые КС, которые предназначены для данного ОУ, хранятся в регистре EP2.

Регистр содержит КС и поля состояния:

- ERROR (ошибки чётности в принятых данных);
- TIMEOUT (произошёл тайм-аут).

Для транзакции формата 5 принятое СД хранится в регистре EP1.

ОУ функционирует одновременно в каналах А и В, вследствие чего отсутствует необходимость в задании рабочего канала. ОУ определяет, откуда приходит КС и отправляет ОС в тот канал, откуда было принято КС.

3.8.8.5 Функционирование контроллера последовательного интерфейса по ГОСТ Р 52070 в режиме монитор

Для функционирования контроллера последовательного интерфейса по ГОСТ Р 52070 в режиме М, необходимо:

- задать время паузы между сообщениями (см. регистр SETTING1);
- задать время ожидания следующего слова (см. регистр SETTING1);
- задать времена 1,25; 1,5; 1,75 и 2,25 мкс (см. регистр SETTING3);
- задать времена 0,5 мкс (см. регистр SETTING2);
- задать поле NOACTTIME (см. регистр SETTING2);
- задать адрес первого байта области памяти, выделенного для монитора (см. регистр MONITOR0);
- задать количество 32-битных слов, выделенных для монитора (см. регистр MONITOR1);
- установить режим работы как монитор (см. регистр SETTING0).

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>					Лист
Изм.	Лист	№ докум.	Подп.	Дата						71

В режиме монитора контроллер принимает все сообщения по интерфейсу и записывает их в память. Каждое 32-битное слово содержит одно принятое сообщение. Формат записи принятых данных показан в таблице 3.8.26.

Таблица 3.8.26 — Формат записи принятых данных

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 19	–	
18	CHANNEL	с какого канала был принято слово; 0 — с канала А; 1 — с канала В;
17	CWORDW	бит принадлежности слова к командному слову или слову данных; 0 — командное, 1 — данные
16	DATAERR	бит ошибки принятых данных
15 – 0	DATA	принятые данные

Монитор принимает независимо по обоим каналам.

3.8.8.6 Инженерные регистры

Инженерные регистры находятся по адресу со смещением 0x100.

Включение режима происходит по записи в регистр по адресу 0x100 слова 0xDEADBEEE.

В регистре по адресу 0x104 задается режим работы. Формат записи принятых данных показан в таблице 3.8.27.

Таблица 3.8.27 — Формат записи принятых данных

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 7	–	
6 – 5	MODE	задает в какое слово вносить ошибку — командное слово или слово данных; 01 — командное слово; 10 — слово данных;
4 – 0	COUNT	задает порядковый номер слова, в которое нужно внести ошибку

3.9 Контроллер интерфейса SpaceWire

3.9.1 Описание функционирования контроллера интерфейса SpaceWire

Структурная схема контроллера интерфейса SpaceWire представлена на рисунке 3.9.1.

Ине. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.					
Изм	Лист	№ докум.	Подп.	Дата					
					Микросхема интегральная 5023BC016				
					Техническое описание				
					Версия 2.04 от 01.09.2017 г.				
					Лист				
					72				

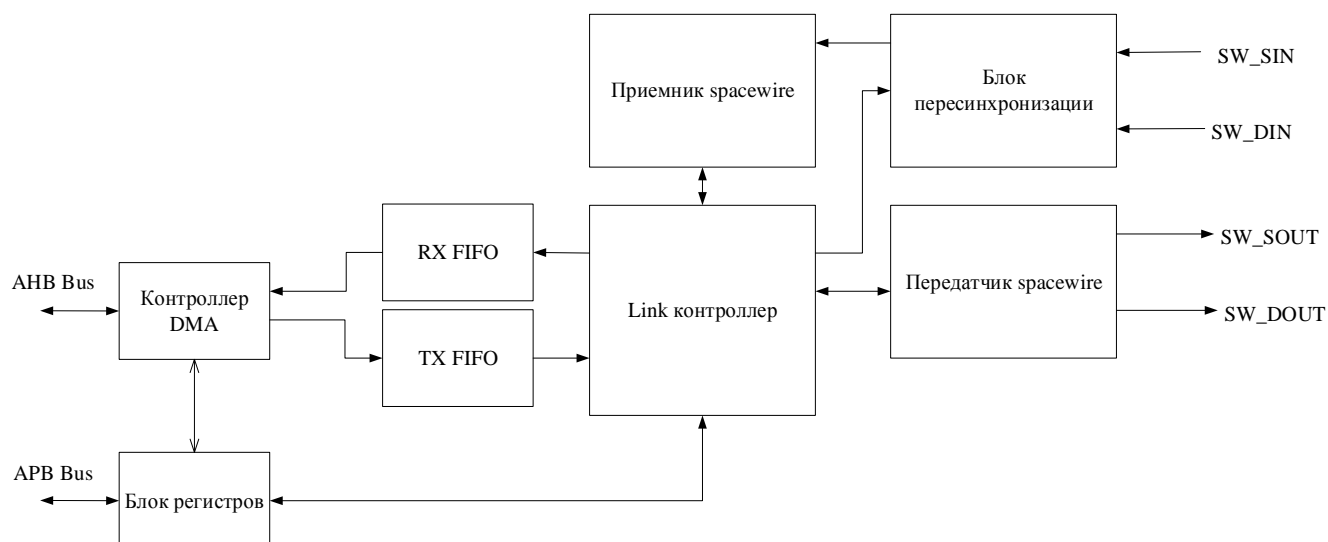


Рисунок 3.9.1 — Структурная схема контроллера интерфейса SpaceWire

Контроллер интерфейса SpaceWire состоит из следующих функциональных блоков:

- Link-контролера;
- контроллера DMA;
- блока регистров;
- блок памяти First in/First out (FIFO) принимаемых данных RX FIFO;
- блок памяти TX FIFO передаваемых данных;
- приёмника SpaceWire;
- передатчика SpaceWire;
- блока пересинхронизации.

3.9.1.1 Link-контроллер

Link-контроллер осуществляет соединение по каналу SpaceWire. Управление Link-контроллером производится битами регистра CONTROL.

При установке бита linkstart в состояние «1» контроллер осуществляет соединение с удаленным интерфейсом. При возникновении ошибки в случае ошибки чётности или разъединении линии контроллер сбрасывается и снова устанавливает соединение, если бит linkstart все еще находится в состоянии «1».

Если бит linkstart равен «0», а бит autostart — «1», то контроллер ожидает начала передачи с другого конца линии и далее выдает сигнал подтверждения.

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	— блок памяти TX FIFO передаваемых данных;
					— приёмника SpaceWire;
					— передатчика SpaceWire;
					— блока пересинхронизации.
					3.9.1.1 Link-контроллер
					Link-контроллер осуществляет соединение по каналу SpaceWire. Управление Link-контроллером производится битами регистра CONTROL.
					При установке бита linkstart в состояние «1» контроллер осуществляет соединение с удаленным интерфейсом. При возникновении ошибки в случае ошибки чётности или разъединении линии контроллер сбрасывается и снова устанавливает соединение, если бит linkstart все еще находится в состоянии «1».
					Если бит linkstart равен «0», а бит autostart — «1», то контроллер ожидает начала передачи с другого конца линии и далее выдает сигнал подтверждения.

Если бит linkdisable равен «1», текущее соединение закрывается, и новое не устанавливается вне зависимости от состояния бита linkstart.

Соединение не может быть установлено, если память RXFIFO заполнена.

Регистр Status отражает текущее состояние Link-контроллера. Если соединение обрывается из-за ошибки, то выставляется соответствующий бит в данном регистре. Для сброса бита ошибки необходимо записать «1» в регистр Status.

3.9.1.2 Приём данных

Принимаемые данные передаются по шине АНВ в ОЗУ СБИС CPU. Блок RXFIFO используется для хранения принимаемых данных, пока блок контроллера DMA не будет готов для передачи данных по шине АНВ. Передача по шине SpaceWire будет производиться до заполнения памяти RXFIFO. В случае возникновения ошибки в середине пакета выставляется бит ошибки в блоке RXFIFO.

Для корректного приёма данных частота системного тактового сигнала должна быть не менее частоты SW_CLK /4.

3.9.1.3 Передача данных

Передача данных из ОЗУ СБИС CPU осуществляется по шине АНВ. Для пересинхронизации данных с тактовым сигналом SpaceWire используется блок TXFIFO. В случае возникновения ошибки соединения при передаче данных выставляется бит EOP.

Передача данных осуществляется по внешнему тактовому сигналу SW_CLK. Скорость передачи задаётся с помощью делителя, коэффициент деления которого записывается в регистр TXScaler. После сброса значение данного регистра равно 10. Инициализация соединения всегда осуществляется на частоте SW_CLK /10.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	<div>Микросхема интегральная 5023BC016</div> <div>Техническое описание</div> <div>Версия 2.04 от 01.09.2017 г.</div>					Лист
Изм	Лист	№ докум.	Подп.	Дата						74

3.9.2 Контроллер BUS MASTER

3.9.2.1 Описание контроллера BUS MASTER

Приём и передача пакетов данных осуществляется по шине АНВ. При этом контроллер SpaceWire является ведущим устройством. Для приёма и передачи данных используется два отдельных канала DMA.

Каждый канал управляется с помощью собственной таблицы дескрипторов.

Таблица дескрипторов расположена в ОЗУ СБИС CPU. Адрес начала таблицы дескрипторов для каждого канала задаётся в соответствующих регистрах контроллера.

Размер таблицы дескрипторов равен 32, при этом размер каждого дескриптора равен 8 байтам. Адрес первого дескриптора в памяти должен быть выровнен относительно 256 байт.

Если канал DMA запущен, последний производит передачу данных до тех пор, пока присутствуют дескрипторы в таблице с активированным битом EN. При достижении конца таблицы процесс продолжается с начала. При установке пользователем бита Wrap контроллер DMA возвращается в начало таблицы и так далее, до достижения ее конца.

Если контроллер DMA считывает дескриптор без активированного бита EN, он завершает обработку дескрипторов в канале. Пользователь может запустить канал снова установкой нового дескриптора.

3.9.2.2 Формат дескриптора

Дескриптор состоит из двух 32-битных слов. В первом содержится размер передаваемых по DMA данных и набор флагов. Второе слово содержит адрес в памяти передаваемых данных. Этот адрес должен быть выровнен относительно 4 байт. Описание бит дескриптора приведено в таблице 3.9.1.

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	<div>Микросхема интегральная 5023BC016</div> <div>Техническое описание</div> <div>Версия 2.04 от 01.09.2017 г.</div>					Лист
Изм.	Лист	№ докум.	Подп.	Дата						75

Таблица 3.9.1 — Описание бит дескриптора

№ бита	Описание
15 – 0	(RX, enable) — количество байт, которые необходимо принять (RX, completed) — количество принятых байт (TX, enable) — количество байт, которые необходимо передать (TX, completed) — неопределено
16	EN: установлено в «1», когда дескриптор разрешено пользователем, но еще не завершен контроллером
17	WR: прерывание списка дескрипторов. При чтении данного бита контроллер начинает выборку данных с начала таблицы
18	IE: разрешение прерывания и выставления статуса по завершении дескриптора
19	DONE: выставляется в «1», когда дескриптор завершен
20	(RX, completed) EOP: выставляется в «1», если получен признак EOP в конце пакета (TX, enable) EOP: выставляется в «1», если необходимо отправить EOP в конце пакета
21	(RX, completed) EEP: выставляется в «1», если получен признак EEP в конце пакета (TX, enable) EEP: выставляется в «1», если необходимо отправить признак EEP в конце пакета

3.9.3 Регистры контроллера интерфейса SpaceWire

3.9.3.1 Состав регистров контроллера интерфейса SpaceWire

Адреса и функциональные назначения регистров контроллера интерфейса SpaceWire приведены в таблице 3.9.2.

Назначения разрядов регистров контроллера интерфейса SpaceWire приведены в таблицах 3.9.3 – 3.9.11.

Таблица 3.9.2 — Адреса и функциональные назначения регистров контроллера интерфейса SpaceWire

Смещение	Название регистра	Тип	Значение по сбросу	Описание
1	2	3	4	5
0x00	CONTROL	RW	0x05000000	Регистр управления
0x04	STATUS	RO	0x00004000	Регистр статуса
0x08	TXSCALER	RW	0x00000009	Регистр делителя клоков
0x0C	TIMECODE	RW	0x00000000	Регистр временных меток
0x10	RXDESCPTR	RW	0x00000000	Регистр указателя на таблицу дескрипторов приёма
0x14	TXDESCPTR	RW	0x00000000	Регистр указателя на таблицу дескрипторов передачи
0x18	TXSCALERLINK	RW	0x00000009	Регистр делителя тактовых сигналов во время Link

Изм.	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.					Лист 76
------	------	----------	-------	------	---	--	--	--	--	------------

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.
--------------	--------------	--------------	--------------	---------------

Адреса и функциональные назначения регистров контроллера интерфейса SpaceWire приведены в таблице 3.9.2.

Назначения разрядов регистров контроллера интерфейса SpaceWire приведены в таблицах 3.9.3 – 3.9.11.

Таблица 3.9.2 — Адреса и функциональные назначения регистров контроллера интерфейса SpaceWire

Смещение	Название регистра	Тип	Значение по сбросу	Описание
1	2	3	4	5
0x00	CONTROL	RW	0x05000000	Регистр управления
0x04	STATUS	RO	0x00004000	Регистр статуса
0x08	TXSCALER	RW	0x00000009	Регистр делителя клоков
0x0C	TIMECODE	RW	0x00000000	Регистр временных меток
0x10	RXDESCPTR	RW	0x00000000	Регистр указателя на таблицу дескрипторов приёма
0x14	TXDESCPTR	RW	0x00000000	Регистр указателя на таблицу дескрипторов передачи
0x18	TXSCALERLINK	RW	0x00000009	Регистр делителя тактовых сигналов во время Link

Продолжение таблицы 3.9.2

1	2	3	4	5
0x1C	RESETTIME	RW	0x00000280	Регистр задания времени ожидания при Link (6,4 мкс) в тактах системной частоты
0x20	DISCONNECTTIME	RW	0x00000055	Регистр задания времени таймаута приёмника (850 нс) (в тактах системной частоты)
0x24	ENDIANNESS	RW	0x00000000	Регистр задания эндианности на приёме и на передаче

3.9.3.2 Регистр CONTROL

Таблица 3.9.3 — Описание регистра CONTROL

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 28	–	
27 – 24	DESCTABLESIZE	Размер таблицы дескрипторов, доступен только на чтение
23 – 24	–	
13	INT_EN_RX_PACKET	Разрешение прерывания при приёме пакета по завершению работы TX дескриптора
12	INT_EN_TX_DSCR_COMPLETE	Разрешение прерывания по завершению TX дескриптора при IE = «1»
11	INT_EN_RX_DSCR_COMPLETE	Разрешение прерывания по завершению RX дескриптора при IE = «1»
10	INT_EN_TIME_CODE	Разрешение прерывания при приёме временных меток
9	INT_EN_LINK_UP_DOWN	Разрешение прерывания при установке соединения
8	CAN_CEL_TX_DMA	Запись «1» прерывает работу TXDMA и RXDMA
7	RESTART_TX_DMA	Запись «1» запускает TXDMA
6	RESTART_RX_DMA	Запись «1» запускает RXDMA
5	ALLOW_TIMECODE_TX	Разрешение передачи временных меток
4	LINK_DISABLE	Linkdisable; запись «1» запрещает работу Link контроллера.
3	LINK_AUTOSTART	Autostart; Запись «1» разрешает автоматическую установку соединения в случае потери.
2	LINK_START	Linkstart; запись «1» разрешает установку соединения.
1	RESET_DMA	Запись «1» сбрасывает контроллер DMA
0	RESET_AMBA	Запись «1» сбрасывает контроллер

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.

Изм	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 77
-----	------	----------	-------	------	---	------------

3.9.3.3 Регистр STATUS

Таблица 3.9.4 — Описание регистра STATUS

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
14	RXFIFO_EMPTY	(только чтение) — память RXFIFO не содержит данных, и последний пакет был отправлен по RXDMA
13	PACKET_RX	Получен пакет. Для сброса бита необходимо записать «1»
12	TX_COMPLETE	TX дескриптор завершен при IE = «1». Для сброса бита необходимо записать «1»
11	RX_COMPLETE	RX дескриптор завершен при IE = «1». Для сброса бита необходимо записать «1»
10	RECEIVED_TIMECODE	Получен пакет с временной меткой. Для сброса бита необходимо записать «1»
9	—	
8	AHB_ERR	Ошибка шины AHB. Для сброса необходимо сбросить контроллер DMA.
7	TX_DMA_RUNNING	TXDMA запущен (только чтение).
6	RX_DMA_RUNNING	RXDMA запущен (только чтение).
5	CREDIT_ERR	Ошибка credit. Для сброса бита необходимо записать «1»
4	ESCAPE_ERR	Ошибка escape. Для сброса бита необходимо записать «1»
3	PARITY_ERR	Ошибка четности. Для сброса бита необходимо записать «1»
2	DISCONNECT_ERR	Ошибка соединения. Для сброса бита необходимо записать «1»
1 – 0	LINK_STATUS	Статус Link контроллера: 0 = выключен, 1 = старт, 2 = соединение установлено, 3 = запущен (только чтение).

3.9.3.4 Регистр TXSCALER

Таблица 3.9.5 — Описание регистра TXSCALER

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
7 – 0	DIVISOR	Делитель частоты тактового сигнала передатчика — 1

3.9.3.5 Регистр RXDESCPTR

Таблица 3.9.6 — Описание регистра RXDESCPTR

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 7	DSCR_ADDR	Адрес начала таблицы дескрипторов
8 – 3	DSCR_NUM	Номер дескриптора
2 – 0	—	Зарезервировано. Должны быть нулевые значения

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	Микросхема интегральная 5023BC016					Лист
					Техническое описание					
					Версия 2.04 от 01.09.2017 г.					78
					Изм.	Лист	№ докум.	Подп.	Дата	

3.9.3.6 Регистр TXDESCPTR

Таблица 3.9.7 — Описание регистра TXDESCPTR

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 8	DSCR_ADDR	Адрес начала таблицы дескрипторов
7 – 3	DSCR_NUM	Номер дескриптора
2 – 0		Зарезервировано. Должны быть нулевые значения

3.9.3.7 Регистр TXSCALERLINK

Таблица 3.9.8 — Описание регистра TXSCALERLINK

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 8		Зарезервировано. Читаются нулевые значения
7 – 0	DIVISOR	TxScalerLink

3.9.3.8 Регистр RESETTIME

Таблица 3.9.9 — Описание регистра RESETTIME

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 11		Зарезервировано. Читаются нулевые значения
10 – 0	RESET_TIME	ResetTime

3.9.3.9 Регистр DISCONNECTTIME

Таблица 3.9.10 — Описание регистра DISCONNECTTIME

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 8		Зарезервировано. Читаются нулевые значения
7 – 0	DISCONNECT_TIME	DisconnectTime

3.9.3.10 Регистр ENDIANNESS

Таблица 3.9.11 — Описание регистра ENDIANNESS

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 2		Зарезервировано. Читаются нулевые значения
1	RX_ENDIAN	RxEndianness. Значение 0 — little endian, 1 — big endian
0	TX_ENDIAN	TxEendianness. Значение 0 — little endian, 1 — big endian

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.					Лист
					Изм	Лист	№ докум.	Подп.	Дата	
										79

3.10 Контроллер прямого доступа к памяти

3.10.1 Общее описание

3.10.1.1 Введение

Контроллер ПДП используется для автоматической передачи данных между интерфейсами SPI и UART и интерфейсами памяти (внутренней либо внешней), либо из памяти в память.

Контроллер ПДП имеет 24 канала, часть из которых жестко привязана к определённым блокам периферии (остальные каналы могут быть использованы только для передачи данных из одной области памяти в другую).

Соответствие каналов контроллера ПДП периферийным блокам приведено в таблице 3.10.1.

Таблица 3.10.1 — Перечень каналов контроллера ПДП

Номера канала	Описание
0	Интерфейс SPI 1 TX
1	Интерфейс SPI 1 RX
2	Интерфейс SPI 2 TX
3	Интерфейс SPI 2 RX
4	Интерфейс UART 1 TX
5	Интерфейс UART 1 RX
6	Интерфейс UART 2 TX
7	Интерфейс UART 2 RX
8	Интерфейс UART 3 TX
9	Интерфейс UART 3 RX
10	Интерфейс UART 4 TX
11	Интерфейс UART 4 RX
12	Интерфейс UART 5 TX
13	Интерфейс UART 5 RX
14	Интерфейс UART 6 TX
15	Интерфейс UART 6 RX
16 – 23	Каналы, не привязанные к блокам периферии

3.10.1.2 Основные характеристики контроллера DMA

Основные характеристики контроллера DMA:

а) 24 канала DMA;

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	Микросхема интегральная 5023BC016					Лист
					Техническое описание					80
					Версия 2.04 от 01.09.2017 г.					
					Изм.	Лист	№ докум.	Подп.	Дата	

- б) каждый канал DMA имеет свои сигналы управления передачей данных;
- в) каждый канал DMA имеет программируемый уровень приоритета;
- г) каждый уровень приоритета обрабатывается, исходя из уровня приоритета, определяемого номером канала DMA;
- д) поддержка различного типа передачи данных:
 - 1) память – память;
 - 2) память – периферия;
 - 3) периферия – память;
- е) поддержка различных типов DMA циклов;
- ж) поддержка передачи данных различной разрядности;
- з) каждому каналу DMA доступна первичная и альтернативная структура управляющих данных;
- и) все управляющие данные канала хранятся в системной памяти;
- к) разрядность данных приёмника равна разрядности данных передатчика;
- л) количество передач в одном цикле DMA может программироваться от 1 до 1024;
- м) инкремент адреса передачи может быть больше чем разрядность данных.

3.10.2 Описание функционирования контроллера DMA

3.10.2.1 Правила арбитража DMA

Контроллер имеет возможность настройки момента арбитража при передачах DMA. Эта возможность позволяет уменьшить время отклика при обслуживании каналов с высоким приоритетом.

Контроллер содержит четыре разряда, которые определяют количество транзакций по шине АНВ до повторения арбитража (перearбитрации). Это значение задаётся параметром R, количество транзакций одного канала до перearбитрации при этом равно 2R.

Таблица 3.10.2 содержит возможную периодичность арбитража в единицах передач по шине АНВ.

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	от 1 до 1024;	
					м) инкремент адреса передачи может быть больше чем разрядность данных.	
					3.10.2 Описание функционирования контроллера DMA	
					3.10.2.1 Правила арбитража DMA	
					Контроллер имеет возможность настройки момента арбитража при передачах DMA. Эта возможность позволяет уменьшить время отклика при обслуживании каналов с высоким приоритетом.	
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	Контроллер содержит четыре разряда, которые определяют количество транзакций по шине АНВ до повторения арбитража (перearбитрации). Это значение задаётся параметром R, количество транзакций одного канала до перearбитрации при этом равно 2R.	
					Таблица 3.10.2 содержит возможную периодичность арбитража в единицах передач по шине АНВ.	
Изм	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 81

Таблица 3.10.2 — Возможная периодичность арбитража в единицах передач по шине АНВ

Значение R	Периодичность переарбитража
b0000	1
b0001	2
b0010	4
b0011	8
b0100	16
b0101	32
b0110	64
b0111	128
b1000	256
b1001	512
b1010 – b1111	1024

Примечание — Необходимо с осторожностью устанавливать большие значения R для низкоприоритетных каналов, так как это может привести к невозможности обслуживать запросы по высокоприоритетным каналам.

При $N > 2R$ (N — номер передачи) и, если результат деления $2R$ на N не целое число, контроллер всегда выполняет последовательность из $2R$ передач до тех пор, пока не станет верным $N < 2R$. Контроллер выполняет оставшиеся N передач в конце цикла DMA.

Значение R задаётся в управляющей структуре канала, описание которой приведено ниже.

3.10.2.2 Приоритет

При проведении арбитража определяется канал для обслуживания в следующем цикле DMA. На выбор следующего канала влияют:

- номер канала;
- уровень приоритета, присвоенного каналу.

Каждому каналу по умолчанию может быть присвоен низкий или высокий уровень приоритета. Присвоение уровня приоритета осуществляется установкой или сбросом разряда `chnl_priority_set`.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	<p>число, контроллер всегда выполняет последовательность из $2R$ передач до тех пор, пока не станет верным $N < 2R$. Контроллер выполняет оставшиеся N передач в конце цикла DMA.</p> <p>Значение R задаётся в управляющей структуре канала, описание которой приведено ниже.</p> <p>3.10.2.2 Приоритет</p> <p>При проведении арбитража определяется канал для обслуживания в следующем цикле DMA. На выбор следующего канала влияют:</p> <ul style="list-style-type: none">— номер канала;— уровень приоритета, присвоенного каналу. <p>Каждому каналу по умолчанию может быть присвоен низкий или высокий уровень приоритета. Присвоение уровня приоритета осуществляется установкой или сбросом разряда <code>chnl_priority_set</code>.</p> <table><tr><td></td><td></td><td></td><td></td><td></td><td rowspan="3"><p>Микросхема интегральная 5023BC016</p><p>Техническое описание</p><p>Версия 2.04 от 01.09.2017 г.</p></td><td rowspan="3">Лист</td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td>Изм.</td><td>Лист</td><td>№ докум.</td><td>Подп.</td><td>Дата</td><td>82</td></tr></table>						<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>	Лист						Изм.	Лист	№ докум.	Подп.	Дата	82
						<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>	Лист																
Изм.	Лист	№ докум.	Подп.	Дата				82															

Канал номер 0 имеет высший уровень приоритета; уровень приоритета снижается с увеличением номера канала.

После окончания цикла DMA для текущего канала контроллер выбирает следующий для обслуживания канал из всех включенных каналов DMA. Рисунок 3.10.1 содержит блок-схему алгоритма выбора следующего канала для обслуживания.

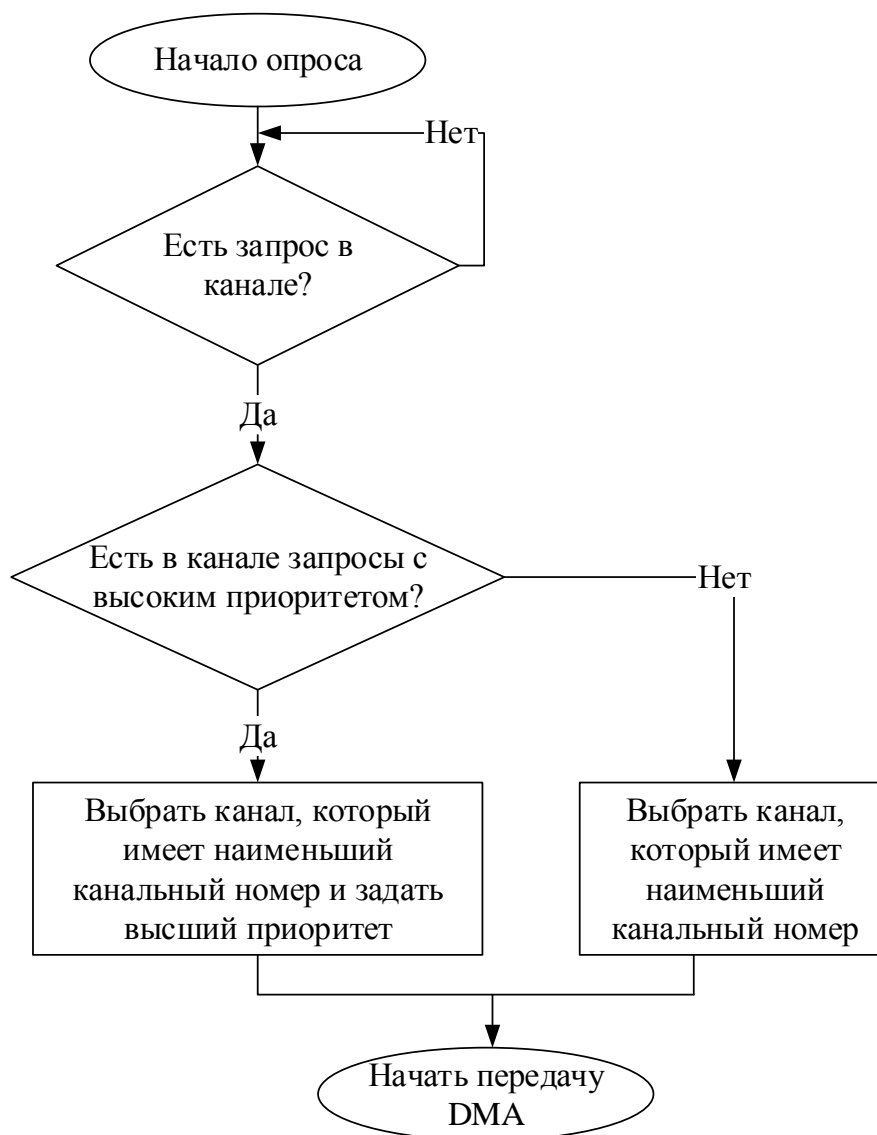


Рисунок 3.10.1 — Алгоритм выбора следующего канала на обслуживание

3.10.2.3 Типы циклов DMA

3.10.2.3.1 Общее описание типов циклов DMA

Разряды `cycle_ctrl` определяют, как контроллер будет выполнять циклы DMA. Описание значений этих разрядов приведено в таблице 3.10.3.

Инв. № подл.	Подп. и дата.				Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 83
	Взам. инв. №					
	Инв. № дубл.					
	Подп. и дата.					
Изм					Копировал	83
Лист						
№ докум.						
Подп.						
Дата						

высоким приоритетом?

Да

Выбрать канал, который имеет наименьший канальный номер и задать высший приоритет

Выбрать канал, который имеет наименьший канальный номер

Начать передачу DMA

Рисунок 3.10.1 — Алгоритм выбора следующего канала на обслуживание

3.10.2.3 Типы циклов DMA

3.10.2.3.1 Общее описание типов циклов DMA

Разряды `cycle_ctrl` определяют, как контроллер будет выполнять циклы DMA. Описание значений этих разрядов приведено в таблице 3.10.3.

Таблица 3.10.3 — Состав циклов DMA

cycle_ctrl	Описание
b000	Структура управляющих данных канала в запрещённом состоянии
b001	Обычный цикл DMA
b010	Авто-запрос
b011	Режим «пинг-понг»
b100	Взаимодействие с памятью в режиме scatter-gather с использованием первичных управляющих данных канала
b101	Взаимодействие с памятью в режиме scatter-gather с использованием альтернативных управляющих данных канала
b110	Взаимодействие с периферией в режиме scatter-gather с использованием первичных управляющих данных канала
b111	Взаимодействие с периферией в режиме scatter-gather с использованием альтернативных управляющих данных канала

Примечание — Разряды cycle_ctrl находятся в области памяти, отведённой под channel_cfg.

Для всех типов циклов DMA повторный арбитраж происходит после 2R передач DMA.

Если установить большой период арбитража на низкоприоритетном канале, то будут заблокированы все запросы на обработку от других каналов до тех пор, пока не будут выполнены 2R передач DMA по текущему каналу. Поэтому, устанавливая значение R, необходимо учитывать, что это может привести к повышенному времени отклика на запрос на обработку от высокоприоритетных каналов.

Предусмотрены следующие типы циклов DMA:

- недействительный;
- основной;
- авто-запрос;
- «пинг-понг»;
- работа с памятью в режиме scatter-gather;
- работа с периферией в режиме scatter-gather.

3.10.2.3.2 Недействительный цикл

После окончания цикла DMA контроллер устанавливает тип цикла «недействительный» для предотвращения повторного выполнения цикла DMA.

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>				Лист
Изм.	Лист	№ докум.	Подп.	Дата					84

3.10.2.3.3 Основной цикл

В режиме «Основной цикл» контроллер функционирует только с основными или альтернативными управляющими данными канала. После того, как разрешено функционирование канала, и контроллер получил запрос на обработку, цикл DMA отрабатывается как показано в таблице 3.10.4.

Таблица 3.10.4 — Описание шагов обработки цикла DMA в режиме «Основной цикл»

Номер шага	Описание
1	Контроллер выполняет 2R передач. Если число оставшихся передач 0, контроллер переходит к шагу 3
2	Осуществление арбитража: — если высокоприоритетный канал выдает запрос на обработку, то контроллер начинает обслуживание этого канала; — если периферийный блок или программное обеспечение выдает запрос на обработку (повторный запрос на обработку по каналу), то контроллер переходит к шагу 1.
3	Контроллер устанавливает прерывание канала DMA. Это указывает центральному процессору на завершение цикла DMA. Данный режим используется для работы с блоками SPI и UART

3.10.2.3.4 Авто-запрос

Функционируя в режиме «Авто-запрос», контроллер ожидает получения одиночного запроса на обработку для разрешения работы и выполнения цикла DMA. Это позволяет выполнять передачу больших пакетов данных без существенного увеличения времени отклика на обслуживание высокоприоритетных запросов и не требует множественных запросов на обработку от процессора или периферийных блоков.

Контроллер позволяет выбрать для использования первичную или альтернативную структуру управляющих данных канала. После того, как разрешено функционирование канала и контроллер получил запрос на обработку, цикл DMA отрабатывается как показано в таблице 3.10.5.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>					Лист
Изм.	Лист	№ докум.	Подп.	Дата						85

Таблица 3.10.5 — Описание шагов обработки цикла DMA в режиме «Авто-запрос»

Номер шага	Описание
1	Контроллер выполняет 2R передач для канала C. Если число оставшихся передач равно нулю, контроллер переходит к шагу 3
2	Осуществление арбитража: — если высокоприоритетный канал выдает запрос на обработку, то контроллер начинает обслуживание этого канала; — если периферийный блок или программное обеспечение выдает запрос на обработку (повторный запрос на обработку по каналу), то контроллер переходит к шагу 1.
3	Контроллер устанавливает прерывание канала DMA, указывает центральному процессору на завершение цикла DMA

Отличие от режима «основной» заключается в том, что в режиме «Авто-запуск» контроллер осуществляет запуск 2R транзакций по одному запросу от блока периферии, в то время как основной режим осуществляет одну транзакцию по соответствующему ему запросу.

3.10.2.3.5 Режим «Пинг-понг»

В режиме «Пинг-понг» контроллер выполняет цикл DMA, используя одну из структур управляющих данных (первичную), а затем выполняет еще один цикл DMA, используя другую структуру управляющих данных (альтернативную).

Контроллер выполняет циклы DMA с переключением структур до тех пор, пока не сосчитает ошибочную структуру данных или пока процессор не запретит работу канала.

Рисунок 3.10.2 содержит диаграмму функционирования контроллера в режиме «пинг-понг».

Шаг А состоит из следующих действий:

- процессор устанавливает первичную структуру управляющих данных для шага «А»;
- процессор устанавливает альтернативную структуру управляющих данных для шага «В». Это позволит контроллеру переключиться к шагу «В» незамедлительно после выполнения шага «А», при условии, что контроллер не получит запрос на обработку от высокоприоритетного канала;
- контроллер получает запрос и выполняет 4 передачи DMA;

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	<p>Контроллер выполняет циклы DMA с переключением структур до тех пор, пока не сосчитает ошибочную структуру данных или пока процессор не запретит работу канала.</p> <p>Рисунок 3.10.2 содержит диаграмму функционирования контроллера в режиме «пинг-понг».</p> <p>Шаг А состоит из следующих действий:</p> <ul style="list-style-type: none">— процессор устанавливает первичную структуру управляющих данных для шага «А»;— процессор устанавливает альтернативную структуру управляющих данных для шага «В». Это позволит контроллеру переключиться к шагу «В» незамедлительно после выполнения шага «А», при условии, что контроллер не получит запрос на обработку от высокоприоритетного канала;— контроллер получает запрос и выполняет 4 передачи DMA;	
Изм.	Лист	№ докум.	Подп.	Дата	<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>	Лист
						86

- контроллер выполняет арбитраж. После получения запроса на обработку от этого же канала, контроллер продолжает цикл в ситуации отсутствия высокоприоритетных запросов;
- контроллер выполняет оставшиеся 2 передачи DMA;
- контроллер устанавливает прерывание канала DMA и входит в процедуру арбитража.

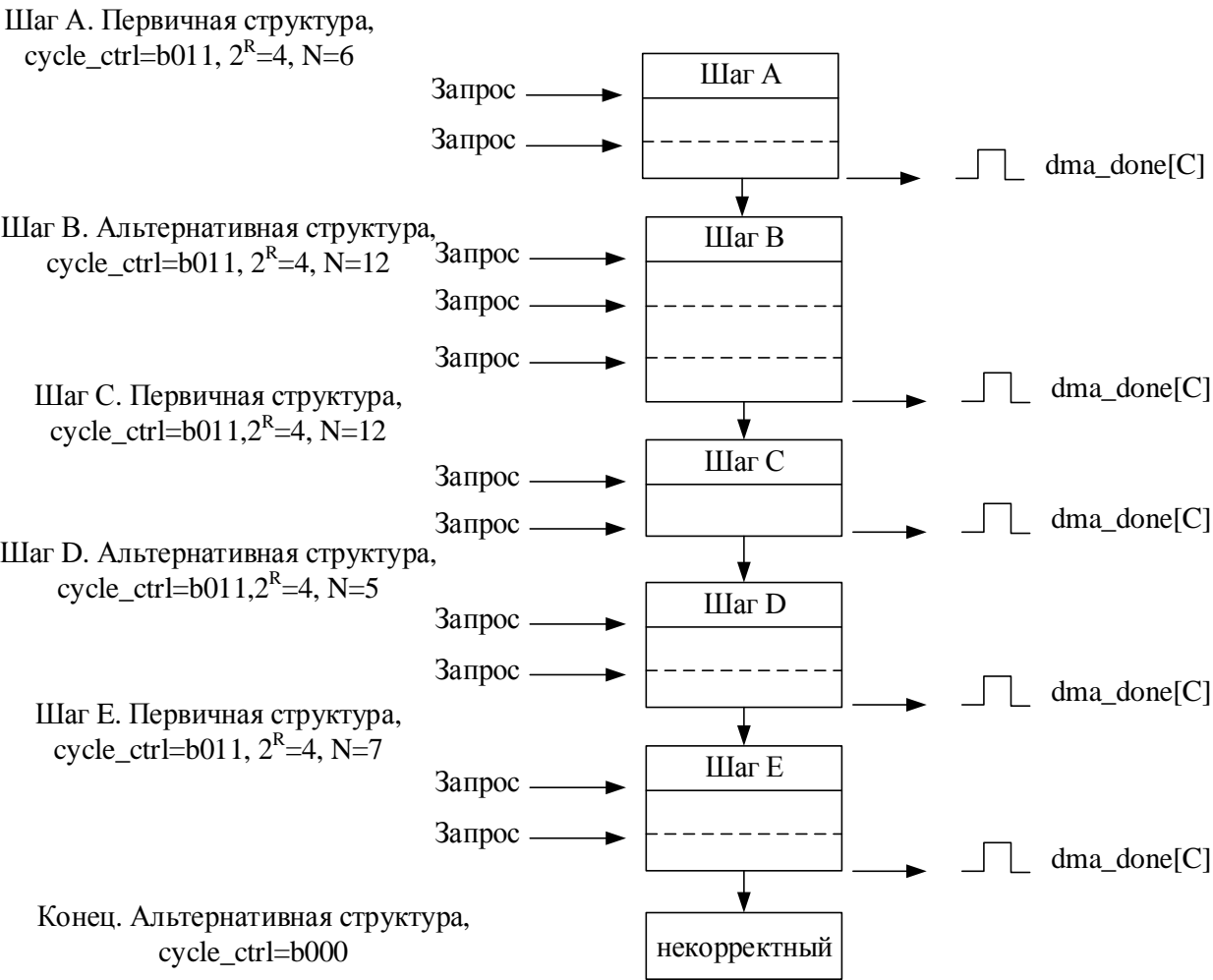


Рисунок 3.10.2 — Диаграмма функционирования контроллера в режиме «пинг-понтг»

После выполнения шага «А» процессор может установить первичные управляющие данные канала для шага «С». Это позволит контроллеру переключиться к шагу «С» незамедлительно после выполнения шага «В», при условии, что контроллер не получит запрос на обработку от высокоприоритетного канала.

После получения нового запроса на обработку от канала при условии его наивысшего приоритета выполняется шаг «В», который состоит из следующих действий:

- контроллер выполняет 4 передачи DMA;
- контроллер выполняет арбитраж. После получения запроса на обработку от этого же канала контроллер продолжает цикл в ситуации отсутствия высокоприоритетных запросов;
- контроллер выполняет 4 передачи DMA;
- контроллер выполняет арбитраж. После получения запроса на обработку от этого же канала контроллер продолжает цикл в ситуации отсутствия высокоприоритетных запросов;
- контроллер выполняет оставшиеся 4 передачи DMA;
- контроллер устанавливает прерывание канала DMA и переходит к процедуре арбитража.

После выполнения шага «В» процессор может установить альтернативные управляющие данные канала для шага «D».

После получения нового запроса на обработку от канала при условии его наивысшего приоритета выполняется шаг «С», который состоит из следующих действий:

- контроллер выполняет две передачи DMA;
- контроллер устанавливает прерывание канала DMA и входит в процедуру арбитража.

После выполнения шага «С» процессор может установить первичные управляющие данные канала для шага «Е».

После получения нового запроса на обработку от канала при условии его наивысшего приоритета выполняется шаг «D», который состоит из следующих действий:

- контроллер выполняет 4 передачи DMA;
- контроллер выполняет арбитраж. После получения запроса на обработку от этого же канала контроллер продолжает цикл в ситуации отсутствия высокоприоритетных запросов;

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.						
Изм.	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.					Лист
										88

- контроллер выполняет оставшуюся передачу DMA;
- контроллер устанавливает прерывание канала DMA и входит в процедуру арбитража.

После получения нового запроса на обработку от канала при условии его наивысшего приоритета выполняется шаг «Е», который состоит из следующих действий:

- контроллер выполняет 4 передачи DMA;
- контроллер выполняет арбитраж. После получения запроса на обработку от этого же канала контроллер продолжает цикл в ситуации отсутствия высокоприоритетных запросов;
- контроллер выполняет оставшиеся 3 передачи DMA;
- контроллер устанавливает прерывание канала DMA и входит в процедуру арбитража.

3.10.2.3.6 Работа с памятью в режиме Scatter-gather

В режиме «Scatter-gather» контроллер, получая начальный запрос на обработку, выполняет четыре передачи DMA, используя первичные управляющие данные. По окончании этих передач контроллер начинает цикл DMA, используя альтернативные управляющие данные. Затем контроллер выполняет дополнительно четыре передачи DMA, используя первичные управляющие данные.

Контроллер продолжает выполнять циклы DMA, меняя структуры управляющих данных, пока не произойдет одно из следующих событий:

- процессор переведёт контроллер в режим «Основной» во время цикла с альтернативной структурой;
- контроллер считает «неправильную» структуру управляющих данных.

Примечание — После исполнения контроллером N передач с использованием первичных управляющих данных, последний аннулирует управляющие путём установки поля cycle_ctrl в 3'b000.

Контроллер устанавливает прерывание канала DMA в этом режиме работы только тогда, когда передача DMA заканчивается с использованием основного цикла.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>					Лист
Изм.	Лист	№ докум.	Подп.	Дата						89

В режиме Scatter-gather контроллер использует первичные управляющие данные для программирования альтернативных управляющих данных.

Таблица 3.10.6 содержит формат заполнения областей памяти channel_cfg, значения которых определяются константами и областей, значения которых определяются пользователем.

Таблица 3.10.6 — Заполнение областей памяти в режиме scatter-gather

№ бита	Обозначение	Значение	Описание
Области с константными значениями			
31 – 30	dst_inc	b'10	Контроллер производит инкремент адреса пословно
29 – 28	dst_size	b'10	Контроллер осуществляет передачу пословно
27 – 26	src_inc	b'10	Контроллер производит инкремент адреса пословно
25 – 24	src_size	b'10	Контроллер осуществляет передачу пословно
17 – 14	R_power	b'0010	Контроллер выполняет 4 передачи DMA
3	next_useburst	b'0	Для данного режима этот разряд должен быть равен 0
2 – 0	cycle_ctrl	b'100	Контроллер функционирует в режиме scatter-gather с периферией
Области со значениями, определяемыми пользователем			
23 – 21	dst_prot_ctrl	–	Определяет состояние HPROT при записи данных в приёмник
20 – 18	src_prot_ctrl	–	Определяет состояние HPROT при чтении данных из источника
13 – 4	N_minus_1	N ¹⁾	Настраивает контроллер на выполнение N передач DMA, где N кратно 4
¹⁾ Так как поле R_power задает значение 4, то необходимо задавать значение N, кратное 4. Число, равное N/4 — это количество итераций, которое нужно настраивать альтернативные управляющие данные.			

Рисунок 3.10.3 содержит диаграмму функционирования в режиме работы scatter-gather с памятью.

Инициализация:

— основной настройкой для активации копий A, B, C и D являются операции:
cycle_ctrl = b100, 2^R = 4, N = 16;

— запись основного источника данных в память осуществляется с помощью структуры, указанной в таблице 3.10.7.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	<div>Микросхема интегральная 5023BC016</div> <div>Техническое описание</div> <div>Версия 2.04 от 01.09.2017 г.</div>					Лист
Изм.	Лист	№ докум.	Подп.	Дата						90

Таблица 3.10.7 — Структура основного источника данных в памяти

Шаг	src_data_end_ptr	dst_data_end_ptr	channel_cfg	Неиспользуемый
A	0x0A000000	0x0AE00000	cycle_ctrl = b101, 2R = 4, N = 3	0xFFFFFFFF
B	0x0B000000	0x0BE00000	cycle_ctrl = b101, 2R = 2, N = 8	0xFFFFFFFF
C	0x0C000000	0x0CE00000	cycle_ctrl = b101, 2R = 8, N = 5	0xFFFFFFFF
D	0x0D000000	0x0DE00000	cycle_ctrl = b101, 2R = 4, N = 4	0xFFFFFFFF

Транзакция памяти в режиме scatter-gather:

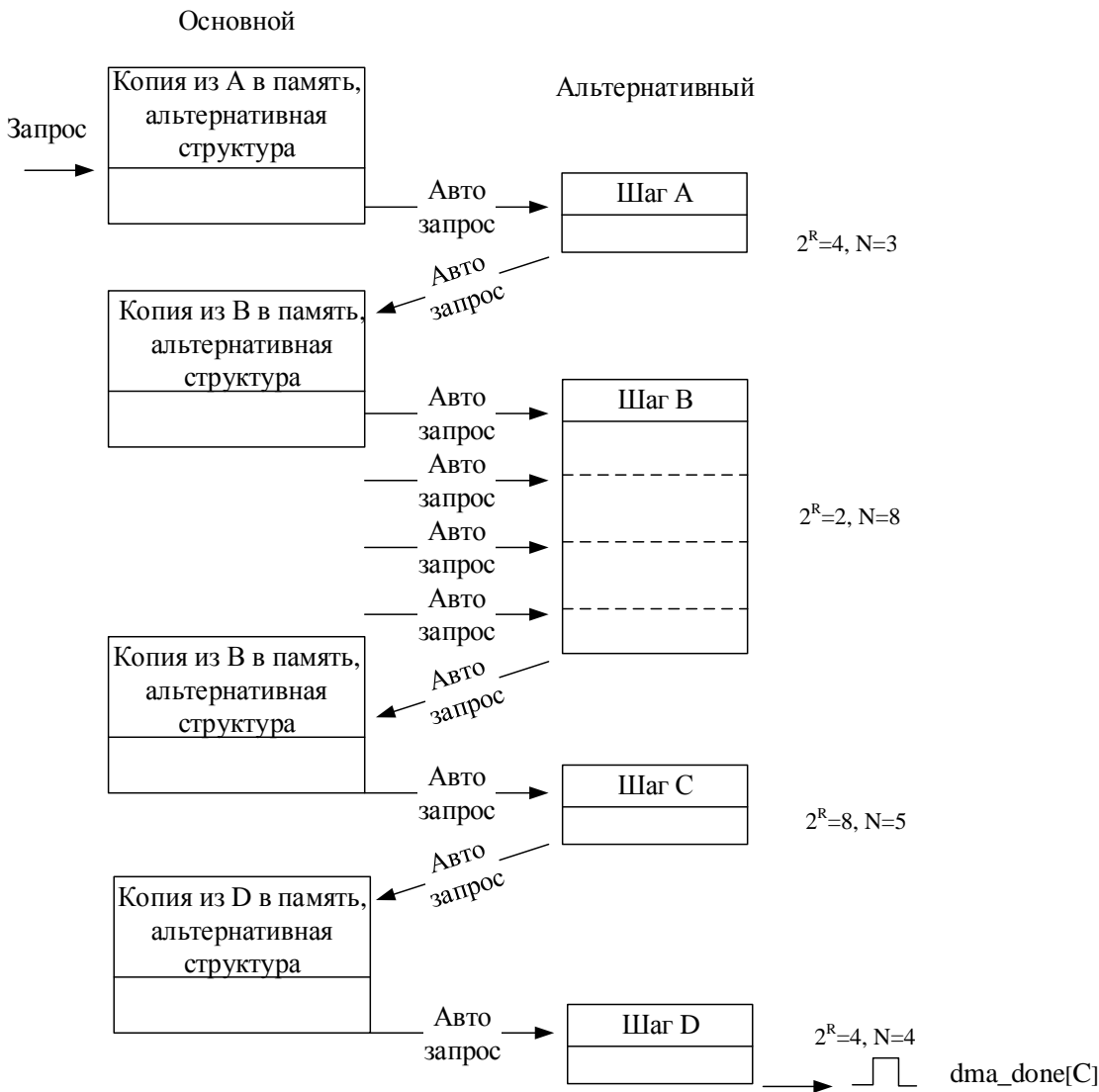


Рисунок 3.10.3 — Пример работы контроллера DMA в режиме scatter-gather с памятью

Инициализация состоит из следующих действий:

— процессор настраивает первичную структуру управляющих данных для работы в режиме работы scatter-gather с памятью путем установки поля cycle_ctrl в

Име. № подл.	Подп. и дата
Взам. инв. №	Име. № дубл.
Подп. и дата	Име. № подл.

Изм.	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016	Лист
					Техническое описание	91
					Версия 2.04 от 01.09.2017 г.	

b100. Так как управляющие данные канала состоят из четырёх слов, необходимо задать 2R равным 4. На рисунке 3.10.3 количество задач равно 4, следовательно, N равно 16;

— процессор записывает управляющие данные для шагов A, B, C, D в область памяти с адресом, указанным в поле src_data_end_ptr;

— процессор разрешает работу канала DMA. Передачи в данном режиме начинают исполняться при получении контроллером запроса на обслуживание по dma_req[] или запроса от процессора.

Порядок выполнения следующий:

— по получению запроса на обслуживание контроллер выполняет четыре передачи DMA. Эти передачи записывают альтернативную структуру управляющих данных для шага «A»;

— контроллер генерирует автозапрос для канала, после чего проводит процедуру арбитража;

— контроллер выполняет шаг «A». По окончании контроллер генерирует автозапрос для канала и проводит процедуру арбитража;

— контроллер выполняет четыре передачи DMA. Эти передачи записывают альтернативную структуру управляющих данных для шага «B»;

— контроллер генерирует автозапрос для канала, после чего проводит процедуру арбитража;

— контроллер выполняет шаг B. По окончании выполнения контроллер генерирует автозапрос для канала и проводит процедуру арбитража;

— контроллер выполняет четыре передачи DMA. Эти передачи записывают альтернативную структуру управляющих данных для шага «C»;

— контроллер генерирует автозапрос для канала, после чего проводит процедуру арбитража;

— контроллер выполняет шаг «C». По окончании контроллер генерирует автозапрос для канала и проводит процедуру арбитража;

— контроллер выполняет четыре передачи DMA. Эти передачи записывают альтернативную структуру управляющих данных для шага «D»;

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	автозапрос для канала и проводит процедуру арбитража;	
					— контроллер выполняет четыре передачи DMA. Эти передачи записывают альтернативную структуру управляющих данных для шага «В»;	
					— контроллер генерирует автозапрос для канала, после чего проводит процедуру арбитража;	
					— контроллер выполняет шаг В. По окончании выполнения контроллер генерирует автозапрос для канала и проводит процедуру арбитража;	
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	— контроллер выполняет четыре передачи DMA. Эти передачи записывают альтернативную структуру управляющих данных для шага «С»;	
					— контроллер генерирует автозапрос для канала, после чего проводит процедуру арбитража;	
					— контроллер выполняет шаг «С». По окончании контроллер генерирует автозапрос для канала и проводит процедуру арбитража;	
					— контроллер выполняет четыре передачи DMA. Эти передачи записывают альтернативную структуру управляющих данных для шага «D»;	
Изм.	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016	Лист
					Техническое описание	
					Версия 2.04 от 01.09.2017 г.	92

— контроллер устанавливает cycle_ctrl первичных управляющих данных в b000 для индикации о том, что эта структура управляющих данных является «неправильной»;

— контроллер генерирует автозапрос для канала, после чего проводит процедуру арбитража;

— контроллер выполняет шаг «D», используя основной цикл DMA;

— контроллер устанавливает прерывание канала DMA и входит в процедуру арбитража.

3.10.2.3.7 Режим работы «Scatter-gather» с периферией

В режиме работы «Scatter-gather» с периферией контроллер, получая начальный запрос на обработку, выполняет четыре передачи DMA, используя первичные управляющие данные. По окончании этих передач контроллер начинает цикл DMA, используя альтернативные управляющие данные без осуществления арбитража и не устанавливая сигнал dma_active[C] в 0.

Примечание — Это единственный случай, при котором контроллер не осуществляет процедуру арбитража после выполнения передачи DMA, используя первичные управляющие данные.

После того, как указанный цикл завершился, контроллер выполняет арбитраж и, при получении запроса на обслуживание от периферии, имеющего наивысший приоритет, он выполняет еще четыре передачи DMA, используя первичные управляющие данные. По окончании этих передач контроллер начинает цикл DMA, используя альтернативные управляющие данные без осуществления арбитража и не устанавливая сигнал dma_active[C] в 0.

Контроллер продолжает выполнять циклы DMA, меняя структуры управляющих данных, пока не произойдет одно из следующих условий:

— процессор переведёт контроллер в режим «Основной» во время цикла с альтернативной структурой;

— контроллер считает «неправильную» структуру управляющих данных.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	<div>Микросхема интегральная 5023BC016</div> <div>Техническое описание</div> <div>Версия 2.04 от 01.09.2017 г.</div>					Лист
Изм	Лист	№ докум.	Подп.	Дата						93

Примечание — После исполнения контроллером N передач с использованием первичных управляющих данных, он делает эти управляющие данные «неправильными» путем установки cycle_ctrl в 3'b000.

Контроллер устанавливает прерывание в режиме работы «Scatter-gather» с периферией только по окончании основного цикла передачи DMA.

В режиме «Scatter-gather» с периферией контроллер использует первичные управляющие данные для программирования альтернативных управляющих данных.

Таблица 3.10.8 содержит формат областей памяти channel_cfg которые должны быть определены константами, и областей, значения которых определяются пользователем.

Таблица 3.10.8 — Заполнение областей памяти в режиме «Scatter-gather» при работе с периферией

№ бита	Обозначение	Значение	Описание
Области с константными значениями			
31 – 30	dst_inc	b'10	Контроллер производит инкремент адреса пословно
29 – 28	dst_size	b'10	Контроллер осуществляет передачу пословно
27 – 26	src_inc	b'10	Контроллер производит инкремент адреса пословно
25 – 24	src_size	b'10	Контроллер осуществляет передачу пословно
17 – 14	R_power	b'0010	Контроллер выполняет 4 передачи DMA
2 – 0	cycle_ctrl	b'100	Контролер функционирует в режиме scatter-gather с периферией
Области со значениями, определяемыми пользователем			
23 – 21	dst_prot_ctrl	—	Определяет состояние HPROT при записи данных в приёмник
20 – 18	src_prot_ctrl	—	Определяет состояние HPROT при чтении данных из источника
13 – 4	N_minus_1	N ¹⁾	Настраивает контроллер на выполнение N передач DMA, где N кратно 4
3	next_useburst	—	При установке в 1 контроллер установит chnl_useburst_set[C] в 1 после выполнения передачи с альтернативной структурой.
¹⁾ Так как поле R_power задает значение 4, то необходимо задавать значение N, кратное 4. Число, равное N/4, это количество итераций настройки альтернативных управляющих данных.			

Име. № подл.	Подп. и дата
Взам. име. №	Име. № дубл.
Подп. и дата	Подп. и дата.

Изм	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 94
-----	------	----------	-------	------	---	------------

Транзакция с периферией в режиме scatter-gather:

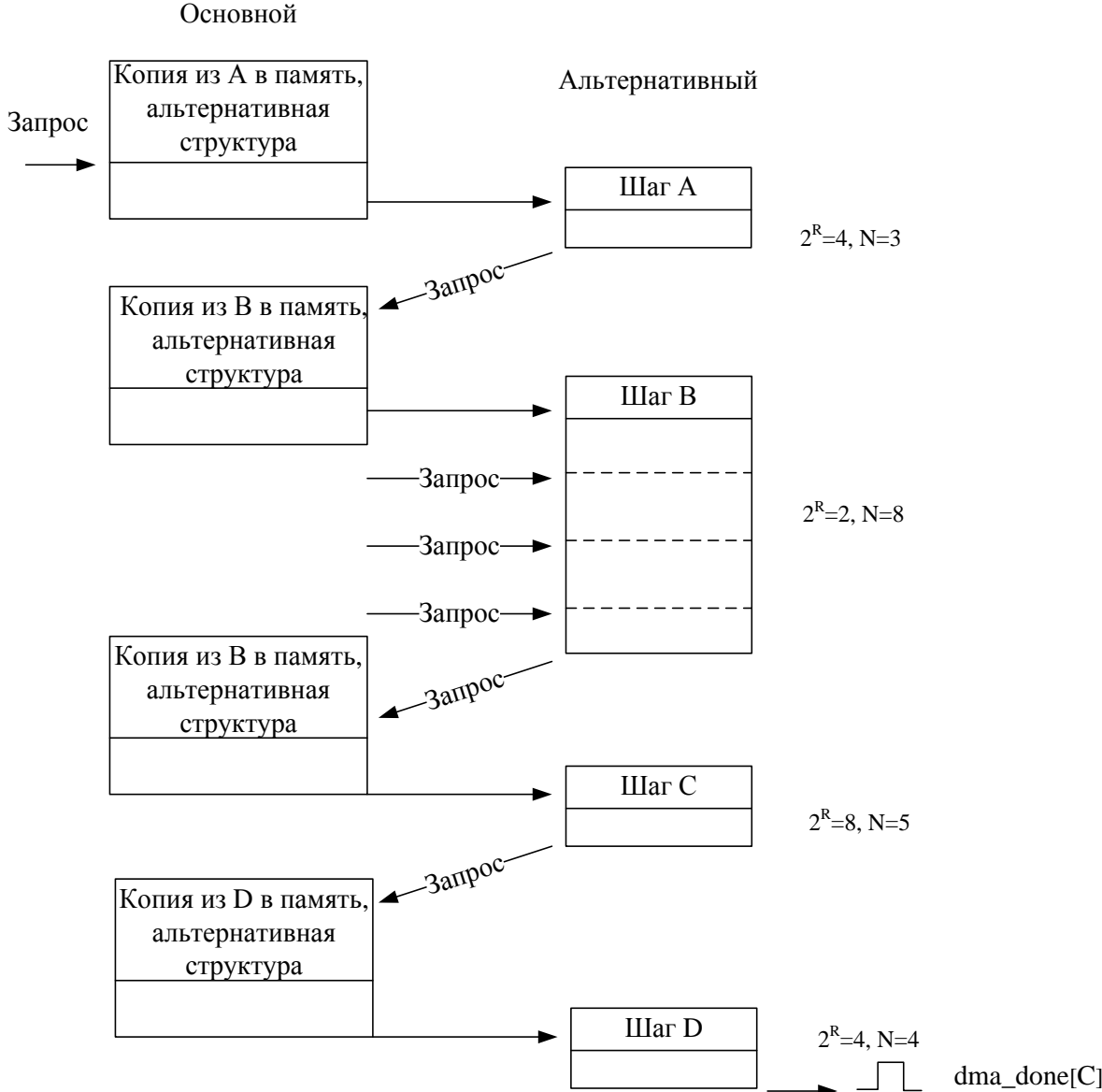


Рисунок 3.10.4 содержит диаграмму функционирования контроллера в режиме «Scatter-gather» с периферией «исполнение с изменением конфигурации».

Инициализация:

- сконфигурировать основную структуру для включения копирования A, B, C и D: cycle_ctrl=b100, $2^R = 4$, $N = 16$;
- запись основного источника данных в память осуществляется с помощью структуры, указанной в таблице 3.10.9.

Ине. № подл.	Подп. и дата.
Взам. инв. №	Инв. № дубл.
Подп. и дата	
Ине. № подл.	

Таблица 3.10.9 — Структура основного источника данных в памяти

Шаг	src_data_end_ptr	dst_data_end_ptr	channel_cfg	Неиспользуемый
A	0x0A000000	0x0AE00000	cycle_ctrl = b111, 2 ^R = 4, N = 3	0xFFFFFFFF
B	0x0B000000	0x0BE00000	cycle_ctrl = b111, 2 ^R = 2, N = 8	0xFFFFFFFF
C	0x0C000000	0x0CE00000	cycle_ctrl = b111, 2 ^R = 8, N = 5	0xFFFFFFFF
D	0x0D000000	0x0DE00000	cycle_ctrl = b001, 2 ^R = 4, N = 4	0xFFFFFFFF

Транзакция с периферией в режиме scatter-gather:

Основной

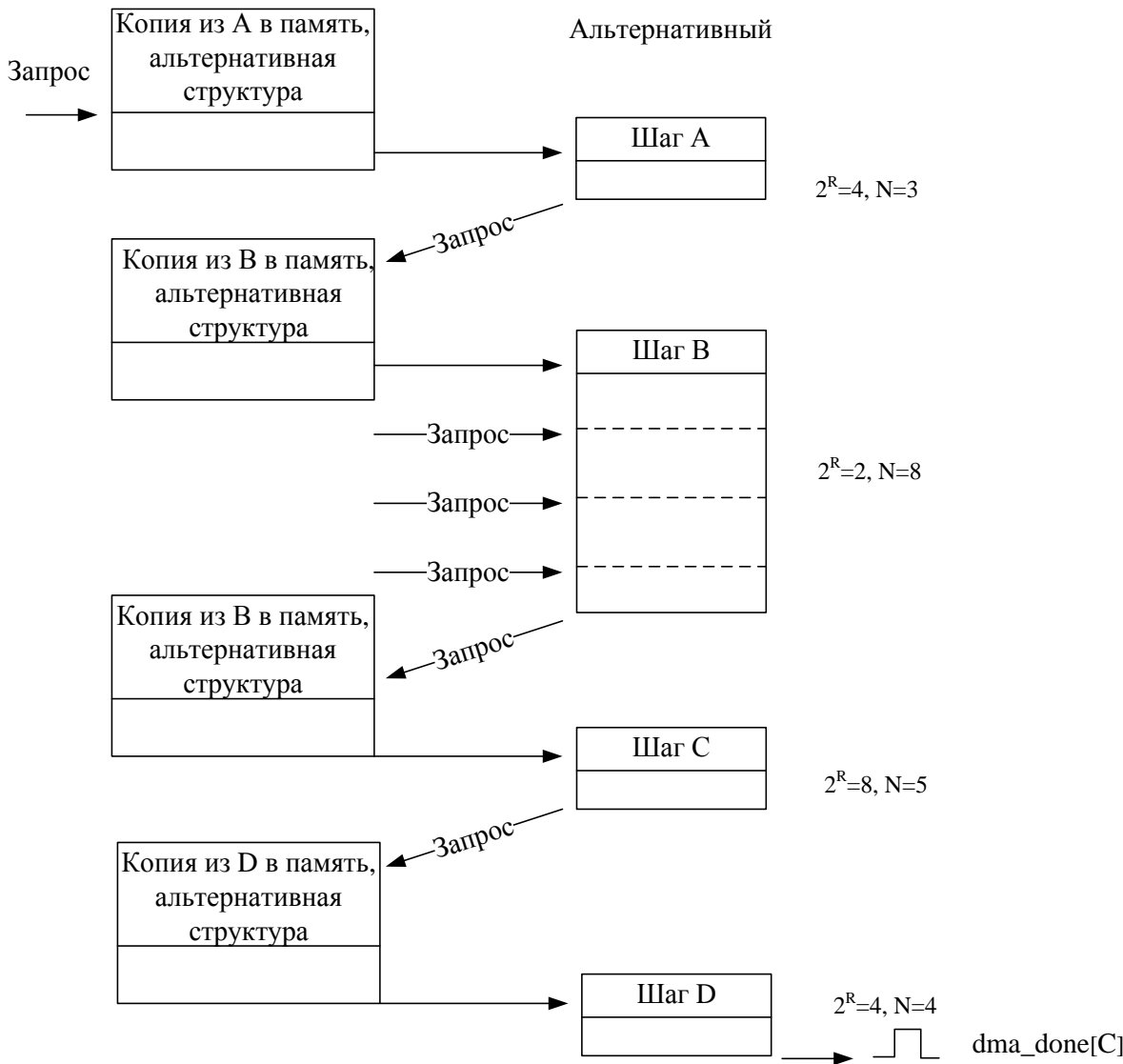


Рисунок 3.10.4 — Диаграмма функционирования контроллера DMA в режиме «Scatter-gather» с периферией

Инициализация состоит в следующем:

— процессор настраивает первичную структуру управляющих данных для работы в режиме «Scatter-gather» с периферией путём установки `cycle_ctrl` в `b110`. Так как управляющие данные канала состоят из 4 слов, необходимо установить `2R` в 4. В этом примере количество задач равно 4 и поэтому `N` установлен в 16;

— процессор записывает управляющие данные для шагов «А», «В», «С», «D» в область памяти с адресом, указанным в поле `src_data_end_ptr`;

— процессор разрешает работу канала DMA.

Передачи в данном режиме начинают исполняться при получении контроллером запроса на обслуживание по `dma_req[]`. Передачи выполняются следующим образом:

— по получению запроса на обслуживание, контроллер выполняет 4 передачи DMA. Эти передачи записывают альтернативную структуру управляющих данных для шага А;

— контроллер выполняет шаг «А»;

— по окончании контроллер проводит процедуру арбитража;

— контроллер выполняет четыре передачи DMA. Эти передачи записывают альтернативную структуру управляющих данных для шага «В»;

— контроллер выполняет шаг «В». Для завершения задачи периферия должна установить последовательно три запроса;

— по окончании контроллер проводит процедуру арбитража;

— контроллер выполняет четыре передачи DMA. Эти передачи записывают альтернативную структуру управляющих данных для шага «С»;

— контроллер выполняет шаг «С»;

— по окончании контроллер проводит процедуру арбитража;

— после выставления периферией нового запроса на обслуживание, при условии, что этот запрос является наиболее приоритетным, процесс продолжается;

— контроллер выполняет четыре передачи DMA. Эти передачи записывают альтернативную структуру управляющих данных для шага «D»;

Ине. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	<div>Микросхема интегральная 5023BC016</div> <div>Техническое описание</div> <div>Версия 2.04 от 01.09.2017 г.</div>					Лист
Изм	Лист	№ докум.	Подп.	Дата						97

- контроллер устанавливает поле `cycle_ctrl` первичных управляющих данных в `b000` для индикации ошибочной структуры данных;
 - контроллер выполняет шаг «D», используя основной цикл DMA;
 - контроллер устанавливает прерывание и входит в процедуру арбитража.
- Индикация ошибок.

При получении контроллером DMA по шине АНВ ответа об ошибке, он выполняет следующие действия:

- отключает канал, связанный с ошибкой;
- устанавливает флаг `dma_err` в состояние «1».

После обнаружения процессором флага `dma_err`, процессор определяет номер канала, который был активен в момент появления ошибки. Для этого он осуществляет следующее:

- чтение регистра `chnl_enable_set` с целью создания списка отключенных каналов. Если канал установил флаг `dma_done[]`, то контроллер отключает канал. Программа, выполняемая процессором, должна всегда хранить данные о каналах, которые недавно установили флаги прерывания;

- процессор должен сравнить список выключенных каналов, полученный в шаге 1, с данными о каналах, которые недавно устанавливали флаги прерывания. Канал, по которому отсутствуют данные об установке флага прерывания, является каналом, с которым связана ошибка.

3.10.3 Программное управление модулем

3.10.3.1 Структура управляющих данных канала

В системной памяти должна быть отведена область для хранения управляющих данных каналов. Системная память должна предоставлять смежную область системной памяти, к которой контроллер и процессор имеют доступ, при этом иметь базовый адрес, который кратен общему размеру структуры управляющих данных канала.

Име. № подл.	Подп. и дата	Взам. име. №	Име. № дубл.	Подп. и дата.	<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>					Лист
Изм.	Лист	№ докум.	Подп.	Дата						98

Рисунок 3.10.5 иллюстрирует область памяти, необходимую контроллеру для структур управляющих данных канала, при использовании всех 32 каналов и опциональной альтернативной структуры управляющих данных.

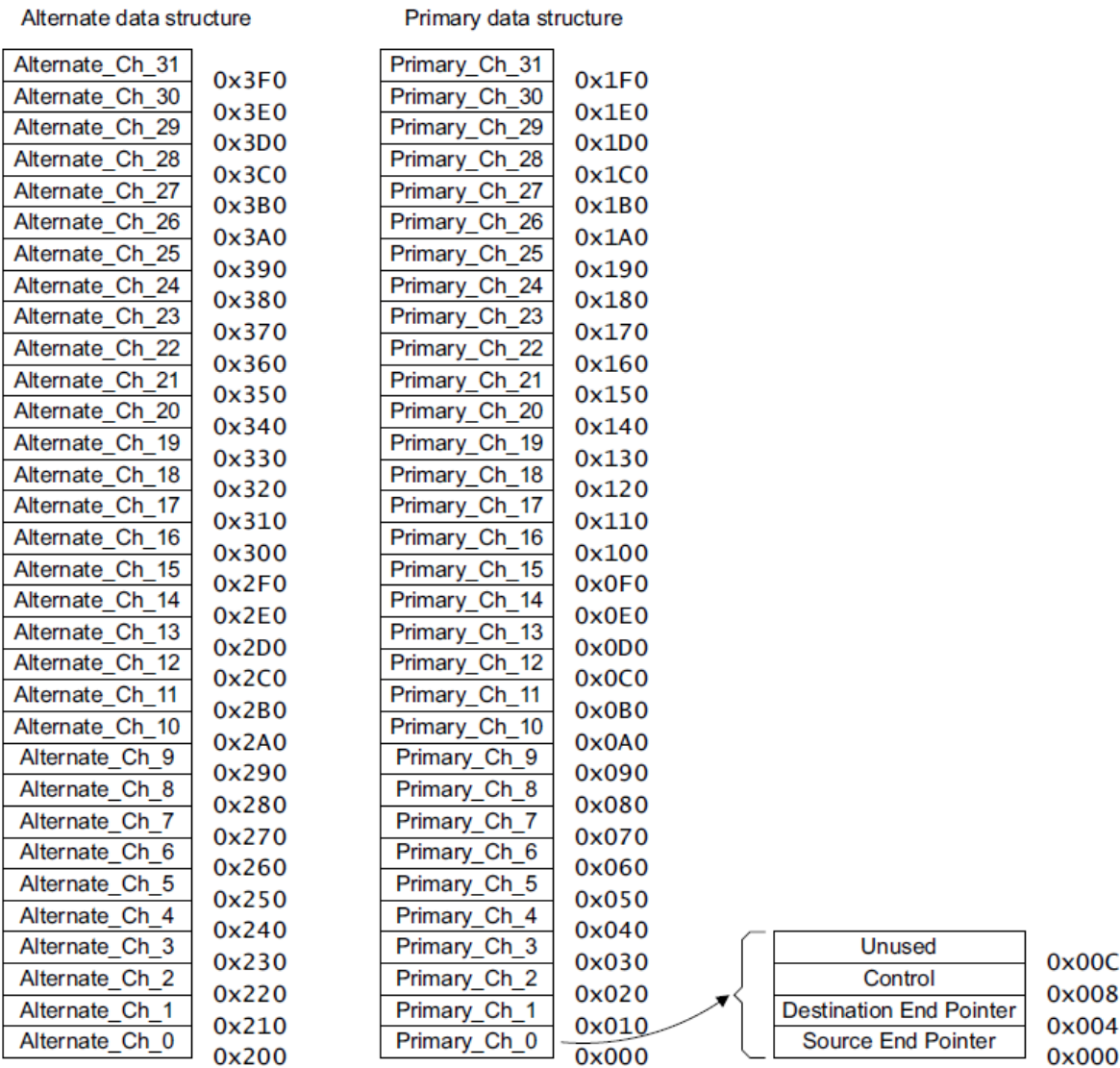


Рисунок 3.10.5 — Карта памяти управляющих данных для 32 каналов памяти

Пример, показанный на рисунке 3.10.5, использует 1 кБ системной памяти. В этом примере контроллер использует младшие 0x10 разрядов адреса для доступа ко всем элементам структуры управляющих данных, и поэтому базовый адрес структуры должен быть 0xXXXXXX000, далее 0xXXXXXX400, далее 0xXXXXXX800, далее 0xXXXXXXC00.

Возможно установить базовый адрес для первичной структуры управляющих данных путём записи соответствующего значения в регистр ctrl_base_ptr.

Необходимый размер области системной памяти зависит от количества каналов, используемых в контроллере, и от того, используется или нет альтернативная структура управляющих данных.

Таблица 3.10.10 содержит формат представления разрядов адреса, обеспечивающих контроллеру DMA доступ к различным элементам структуры управляющих данных, в зависимости от количества каналов, используемых в контроллере.

Таблица 3.10.10 — Формат представления разрядов адреса, обеспечивающих контроллеру DMA доступ к различным элементам структуры управляющих данных

Количество каналов, используемых в контроллере	Разряды адреса						
	[9]	[8]	[7]	[6]	[5]	[4]	[3:0]
1						A	0x0 0x4 0x8
2					A	C[0]	
3 – 4				A	C[1]	C[0]	
5 – 8			A	C[2]	C[1]	C[0]	
9 – 16		A	C[3]	C[2]	C[1]	C[0]	
17 – 32	A	C[4]	C[3]	C[2]	C[1]	C[0]	

Значение A задает одну из структур управляющих данных канала:

- при A = 0 используется первичная структура управляющих данных;
- при A = 1 используется альтернативная структура управляющих данных.

Значения C[x:0] определяют выбор канала DMA.

Address[3:0] задает один из управляющих элементов:

- 0x0 — указатель конца данных источника;
- 0x4 — указатель конца данных приёмника;
- 0x8 — конфигурация управляющих данных;
- 0xC — адрес, недоступный контроллеру.

При необходимости возможно разрешить процессору использование указанных адресов в качестве системной памяти.

Примечание — Нет необходимости в вычислении базового адреса альтернативной структуры управляющих данных, так как регистр alt_ctrl_base_ptr содержит эту информацию.

Ине. № подл.	Подп. и дата.
Взам. инв. №	Ине. № дубл.
Подп. и дата	
Ине. № подл.	

					<i>Микросхема интегральная 5023BC016</i> <i>Техническое описание</i> <i>Версия 2.04 от 01.09.2017 г.</i>	<i>Лист</i>
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подп.</i>	<i>Дата</i>		<i>100</i>

Рисунок 3.10.6 демонстрирует пример реализации контроллера с использованием трёх каналов DMA и с альтернативной структурой управляющих данных.

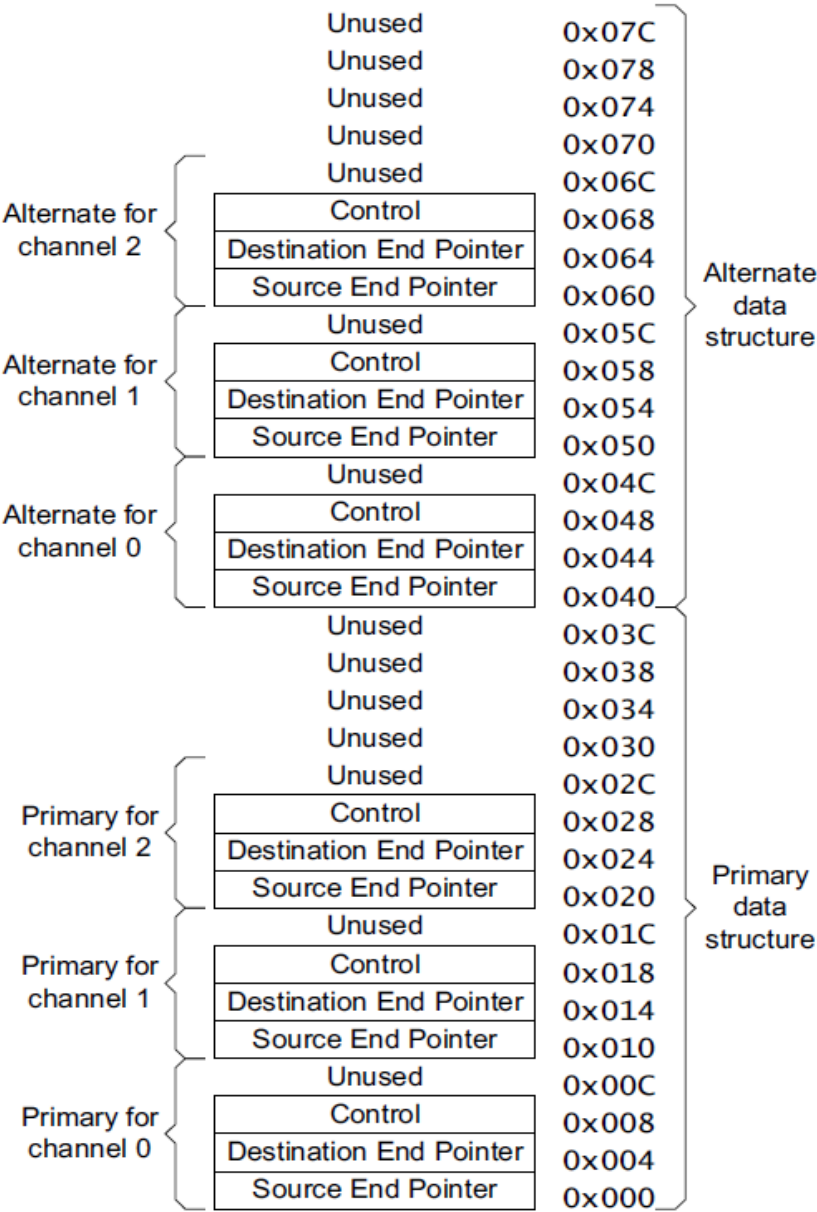


Рисунок 3.10.6 — Карта памяти для трех каналов DMA, включая альтернативную структуру: Destination end pointer — указатель конца данных приёмника; Source end pointer — указатель конца данных источника; Control — управление

В указанном выше примере используется 128 байт системной памяти. Контроллер DMA использует младшие 0x06 разрядов адреса для доступа ко всем элементам структуры управляющих данных. Следовательно, базовый адрес структуры должен быть 0xXXXXXX00, далее 0xXXXXXX80.

Контроллер использует системную память для доступа к двум указателям адреса конца данных и разрядам управления каждого канала. Эти 32-х разрядные области памяти и процедура вычисления контроллером адреса передачи DMA описывают следующие подразделы:

- указатель конца данных источника;
- указатель конца данных приёмника;
- разряды управления;
- вычисление адреса.

3.10.3.2 Указатель конца данных источника

Область памяти `src_data_end_ptr` содержит указатель на последний адрес месторасположения данных источника. Формат указателя конца данных источника содержится в таблице 3.10.11.

Таблица 3.10.11 — Указатель конца данных источника

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31 – 0	<code>src_data_end_ptr</code>	Указатель последнего адреса данных источника

Перед тем, как контроллер выполнит передачу DMA, необходимо определить область памяти, отводимую под указатель конца данных источника. Контроллер DMA считывает значение этой области перед началом 2R передач.

Примечание — Контроллер не имеет доступа по записи в эту область памяти.

3.10.3.3 Указатель конца данных приёмника

Область памяти `dst_data_end_ptr` содержит указатель на последний адрес данных приёмника.

Таблица 3.10.12 содержит формат указателя конца данных приёмника.

Таблица 3.10.12 — Указатель конца данных приёмника

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31 – 0	<code>dst_data_end_ptr</code>	Указатель на последний адрес данных приёмника

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>	Лист
Изм	Лист	№ докум.	Подп.	Дата		102

Перед тем, как контроллер выполнит передачу DMA, необходимо определить область памяти, в которой содержится последний адрес данных приёмника. Контроллер DMA считывает значение этой области перед началом 2R передачи.

Примечание — Контроллер не имеет доступа для записи в указанную выше область памяти.

3.10.3.4 Разряды управления

Область памяти под названием channel_cfg обеспечивает управление каждой передачей DMA.

Наименования разрядов области памяти channel_cfg содержатся в таблице 3.10.13.

Таблица 3.10.13 — Название разрядов области памяти channel_cfg

№ бита	31	30	29	28	27	26	25	24	23 – 21	20 – 18	17 – 14	13 – 4	3	2 – 0
Доступ														
Сброс														
	dst_inc		dst_size		src_inc	dst_prot_ctrl	src_prot_ctrl				R_power	next_useburst	cycle_ctrl	

Назначение разрядов этой области памяти содержится в таблице 3.10.14.

Таблица 3.10.14 — Назначение разрядов области памяти channel_cfg

№ бита	Наименование	Описание
1	2	3
31 – 30	dst_src	<p>Шаг инкремента адреса приёмника.</p> <p>Шаг инкремента адреса зависит от разрядности данных источника.</p> <p>Разрядность данных источника = байт:</p> <p>b00 = байт;</p> <p>b01 = полуслово (16 разрядов);</p> <p>b10 = слово (32 разряда);</p> <p>b11 = нет инкремента. Адрес остается равным значению области памяти dst_data_end_ptr.</p> <p>Разрядность данных источника = полуслово:</p> <p>b00 = зарезервировано;</p> <p>b01 = полуслово;</p> <p>b10 = слово;</p> <p>b11 = нет инкремента. Адрес остается равным значению области памяти dst_data_end_ptr.</p> <p>Разрядность данных источника = слово:</p> <p>b00 = зарезервировано;</p> <p>b01 = зарезервировано;</p> <p>b10 = слово (32 разряда);</p> <p>b11 = нет инкремента. Адрес остается равным значению области памяти dst_data_end_ptr</p>
29 – 28	dst_size	<p>Размерность данных приёмника</p> <p>Примечание:</p> <p>Значение этого поля должно быть равно значению поля src_size.</p>
27 – 26	src_inc	<p>Шаг инкремента адреса источника.</p> <p>Шаг инкремента адреса зависит от разрядности данных источника.</p> <p>Разрядность данных источника = байт:</p> <p>b00 = байт;</p> <p>b01 = полуслово;</p> <p>b10 = слово (32 разряда);</p> <p>b11 = нет инкремента. Адрес остается равным значению области памяти src_data_end_ptr.</p> <p>Разрядность данных источника = полуслово:</p> <p>b00 = зарезервировано</p> <p>b01 = полуслово</p> <p>b10 = слово</p> <p>b11 = нет инкремента. Адрес остается равным значению области памяти src_data_end_ptr.</p> <p>Разрядность данных источника = слово:</p> <p>b00 = зарезервировано;</p> <p>b01 = зарезервировано;</p> <p>b10 = слово;</p> <p>b11 = нет инкремента. Адрес остается равным значению области памяти src_data_end_ptr</p>

Ине. № подл.	Подп. и дата	Взам. инв. №	Ине. № дубл.	Подп. и дата.
Изм	Лист	№ докум.	Подп.	Дата
Микросхема интегральная 5023BC016				Лист
Техническое описание				104
Версия 2.04 от 01.09.2017 г.				

Продолжение таблицы 3.10.14

1	2	3
25 – 24	src_size	<p>Задаёт размерность данных источника:</p> <p>b00 = байт;</p> <p>b01 = полуслово (в русском обычно слово);</p> <p>b10 = слово (в русском обычно двойное слово);</p> <p>b11 = зарезервировано</p>
23 – 21	dst_prot_ctrl	<p>Задаёт состояние HPROT[3:1], когда контроллер записывает данные в приёмник.</p> <p>Разряд 23 управляет разрядом HPROT[3]:</p> <p>0 = HPROT[3] в состоянии «0» и доступ не кэшируется;</p> <p>1 = HPROT[3] в состоянии «1» и доступ кэшируется.</p> <p>Разряд 22 управляет разрядом HPROT[2]:</p> <p>0 = HPROT[2] в состоянии «0» и доступ не буферизуется;</p> <p>1 = HPROT[2] в состоянии «1» и доступ буферизуется.</p> <p>Разряд 21 управляет разрядом HPROT[1]:</p> <p>0 = HPROT[1] в состоянии «0» и доступ непривилегированный;</p> <p>1 = HPROT[1] в состоянии «1» и доступ привилегированный</p>
20 – 18	src_prot_ctrl	<p>Задаёт состояние HPROT[3:1], когда контроллер считывает данные из источника.</p> <p>Разряд 20 управляет разрядом HPROT[3]:</p> <p>0 = HPROT[3] в состоянии 0 и доступ не кэшируется;</p> <p>1 = HPROT[3] в состоянии «1» и доступ кэшируется.</p> <p>Разряд 19 управляет разрядом HPROT[2]:</p> <p>0 = HPROT[2] в состоянии «0» и доступ не буферизуется;</p> <p>1 = HPROT[2] в состоянии «1» и доступ буферизуется.</p> <p>Разряд 18 управляет разрядом HPROT[1]:</p> <p>0 = HPROT[1] в состоянии «0» и доступ непривилегированный;</p> <p>1 = HPROT[1] в состоянии «1» и доступ привилегированный</p>
17 – 14	R_power	<p>Задаёт количество передач DMA до выполнения контроллером процедуры арбитража.</p> <p>Возможные значения:</p> <p>b0000 — арбитраж производится после каждой передачи DMA;</p> <p>b0001 — арбитраж производится после двух передач DMA;</p> <p>b0010 — арбитраж производится после четырёх передач DMA;</p> <p>b0011 — арбитраж производится после восьми передач DMA;</p> <p>b0100 — арбитраж производится после 16 передач DMA;</p> <p>b0101 — арбитраж производится после 32 передач DMA;</p> <p>b0110 — арбитраж производится после 64 передач DMA;</p> <p>b0111 — арбитраж производится после 128 передач DMA;</p> <p>b1000 — арбитраж производится после 256 передач DMA;</p> <p>b1001 — арбитраж производится после 512 передач DMA;</p> <p>b1010 — b1111 - арбитраж производится после 1024 передач DMA. Арбитраж не производится, так как максимальное количество передач DMA равно 1024</p>

Ине. № подл.	Подп. и дата.
Взам. инв. №	Инв. № дубл.
Подп. и дата	
Ине. № подл.	

Изм	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016	Лист
					Техническое описание	
					Версия 2.04 от 01.09.2017 г.	105

Продолжение таблицы 3.10.14

1	2	3
13 – 4	n_minus_1	<p>Перед выполнением цикла DMA эти разряды указывают общее количество передач DMA, из которых состоит цикл DMA. Необходимо установить эти разряды в значение, соответствующее размеру желаемого цикла DMA. 10-разрядное число плюс 1 задает количество передач DMA. Возможные значения:</p> <p>b0000000000 = 1 передача DMA; b0000000001 = 2 передачи DMA; b0000000010 = 3 передачи DMA; b0000000011 = 4 передачи DMA; b0000000100 = 5 передач DMA; b0000000101 = 6 передач DMA; b1111111111 = 1024 передачи DMA.</p> <p>Контроллер обновит это поле перед тем, как произвести процесс арбитража. Это позволяет контроллеру хранить количество оставшихся передач DMA до завершения цикла DMA</p>
3	next_useburst	<p>Контролирует, не установлен ли chnl_useburst_set[C] в состояние «1»; если контроллер функционирует в режиме scatter-gather с периферией и, если контроллер завершает цикл DMA, используя альтернативные управляющие данные.</p> <p>Примечание:</p> <p>Перед завершением цикла DMA, использующего альтернативные управляющие данные, контроллер устанавливает chnl_useburst_set[C] в значение «0», если количество оставшихся передач DMA меньше, чем 2R. Установка next_useburst разряда определяет, будет ли контроллер дополнительно переопределять разряд chnl_useburst_set[C].</p> <p>Если контроллер выполняет цикл DMA в режиме работы scatter-gather с периферией, то после окончания цикла, использующего альтернативные управляющие данные, происходит следующее в зависимости от состояния next_useburst:</p> <p>0 — контроллер не изменяет значение chnl_useburst_set[C]. Если chnl_useburst_set[C] установлен в «0», то для всех оставшихся циклов DMA в режиме работы с периферией «Исполнение с изменением конфигурации», контроллер отвечает на запросы по dma_req[] и dma_sreq[], при выполнении циклов DMA он использует альтернативные управляющие данные.</p> <p>1 — контроллер изменяет значение chnl_useburst_set[C] в состояние «1». Поэтому для оставшихся циклов DMA в режиме работы с периферией «Исполнение с изменением конфигурации», контроллер реагирует только на запросы по dma_req[], при выполнении циклов DMA он использует альтернативные управляющие данные.</p>

Ине. № подл.	Подп. и дата	Взам. инв. №	Ине. № дубл.	Подп. и дата.

Изм	Лист	№ докум.	Подп.	Дата

Микросхема интегральная 5023BC016

Техническое описание

Версия 2.04 от 01.09.2017 г.

Лист

106

Копировал

Формат А4

Продолжение таблицы 3.10.14

1	2	3
2 – 0	cycle_ctrl	<p>Режим работы при выполнении цикла DMA: b000 Стоп. Означает, что структура управляющих данных является «неправильной»; b001 Основной. Контроллер должен получить новый запрос для окончания цикла DMA, перед этим он должен выполнить процедуру арбитража; b010 Авто-запрос. Контроллер автоматически осуществляет запрос на обработку по соответствующему каналу в течение процедуры арбитража. Это означает, что начального запроса на обработку достаточно для выполнения цикла DMA;</p> <p>b011 Пинг-понг. Контроллер выполняет цикл DMA используя одну из структур управляющих данных. По окончании выполнения цикла DMA, контроллер выполняет следующий цикл DMA, используя другую структуру. Контроллер сигнализирует об окончании каждого цикла DMA, позволяя процессору перенастраивать неактивную структуру данных. Контроллер продолжает выполнять циклы DMA, до тех пор, пока он не прочитает «неправильную» структуру данных или пока процессор не изменит cycle_ctrl поле в состояние b001 или b 010;</p> <p>b100 Режим работы scatter-gather с памятью. Смотрите соответствующий раздел.</p> <p>При работе контроллера в данном режиме значение этого поля в первичной структуре управляющих данных должно быть b100;</p> <p>b101 Режим работы scatter-gather с памятью. Смотрите соответствующий раздел. При работе контроллера в данном режиме значение этого поля в альтернативной структуре управляющих данных должно быть b101;</p> <p>b110 Режим работы scatter-gather с периферией. Смотрите соответствующий раздел.</p> <p>При работе контроллера в данном режиме значение этого поля в первичной структуре управляющих данных должно быть b110;</p> <p>b111 Режим работы scatter-gather с периферией. Смотрите соответствующий раздел.</p> <p>При работе контроллера в данном режиме значение этого поля в альтернативной структуре управляющих данных должно быть b111</p>

В начале цикла DMA или 2R передачи DMA контроллер считывает значение channel_cfg из системной памяти. После выполнения 2R или N передач он сохраняет обновлённое значение channel_cfg в системную память.

Контроллер не поддерживает значений dst_size, отличных от значений src_size. Если контроллер обнаруживает неравные значения этих полей, он использует значение src_size в качестве размера данных и приёмника, и источника и при ближайшем обновлении поля n_minus_1, он также устанавливает значение поля dst_size, равное src_size.

Изм.	Лист	№ докум.	Подп.	Дата	<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>	Лист 107

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	<p>режиме значение этого поля в альтернативной структуре управляющих данных должно быть b101; b110 Режим работы scatter-gather с периферией.</p> <p>Смотрите соответствующий раздел.</p> <p>При работе контроллера в данном режиме значение этого поля в первичной структуре управляющих данных должно быть b110; b111 Режим работы scatter-gather с периферией.</p> <p>Смотрите соответствующий раздел.</p> <p>При работе контроллера в данном режиме значение этого поля в альтернативной структуре управляющих данных должно быть b111</p>

В начале цикла DMA или 2R передачи DMA контроллер считывает значение channel_cfg из системной памяти. После выполнения 2R или N передач он сохраняет обновлённое значение channel_cfg в системную память.

Контроллер не поддерживает значений dst_size, отличных от значений src_size. Если контроллер обнаруживает неравные значения этих полей, он использует значение src_size в качестве размера данных и приёмника, и источника и при ближайшем обновлении поля n_minus_1, он также устанавливает значение поля dst_size, равное src_size.

После выполнения контроллером N передач, контроллер устанавливает значение поля `cycle_ctrl` в `b000`, делая тем самым `channel_cfg` данные «неправильными». Это позволяет избежать повторения выполненной передачи DMA.

3.10.3.5 Вычисление адреса

Для вычисления адреса источника передачи DMA, контроллер выполняет сдвиг влево значения `n_minus_1` на количество разрядов, соответствующее полю `src_inc`, и затем вычитает получившееся значение от значения указателя адреса конца данных источника. Подобным образом вычисляется адрес для передачи DMA, контроллер выполняет сдвиг влево значения `n_minus_1` на количество разрядов, соответствующее полю `dst_inc`, и затем вычитает получившееся значение от значения указателя адреса конца данных приёмника. В зависимости от значения полей `src_inc` и `dst_inc` вычисление адресов приёмника и источника выполняется в соответствии со следующими уравнениями:

$$src_inc = b00 \text{ and } dst_inc = b00$$

- адресисточника = $src_data_end_ptr - n_minus_1$
- адресприёмника = $dst_data_end_ptr - n_minus_1$.

$$src_inc = b01 \text{ and } dst_inc = b01$$

- адресисточника = $src_data_end_ptr - (n_minus_1 \ll 1)$
- адресприёмника = $dst_data_end_ptr - (n_minus_1 \ll 1)$.

$$src_inc = b01 \text{ and } dst_inc = b10$$

- адресисточника = $src_data_end_ptr - (n_minus_1 \ll 2)$
- адресприёмника = $dst_data_end_ptr - (n_minus_1 \ll 2)$.

$$src_inc = b11 \text{ and } dst_inc = b11$$

- адрес источника = $src_data_end_ptr$
- адрес приёмника = $dst_data_end_ptr$.

Таблица 3.10.15 содержит перечень адресов приёмника цикла DMA для шести слов с пословным инкрементом.

Име. № подл.	Подп. и дата	Взам. име. №	Име. № дубл.	Подп. и дата.	<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>					Лист
Изм.	Лист	№ докум.	Подп.	Дата						108

Таблица 3.10.15 — Адреса приёмника цикла DMA для шести слов с пословным инкрементом

Начальные значения channel_cfg перед циклом DMA				
src_size=b10, dst_inc=b10, n_minus_1=b101, cycle_ctrl=1				
DMA передачи	Указатель конца данных	Счётчик	Отличие ¹⁾	Адрес
	0x2AC	5	0x14	0x298
	0x2AC	4	0x10	0x29C
	0x2AC	3	0xC	0x2A0
	0x2AC	2	0x8	0x2A4
	0x2AC	1	0x4	0x2A8
	0x2AC	0	0x0	0x2AC
Конечные значения channel_cfg после цикла DMA				
src_size=b10, dst_inc=b10, n_minus_1=0, cycle_ctrl=0				
¹⁾ Это значение, полученное после сдвига влево значения счётчика на количество разрядов, соответствующее dst_inc.				

Таблица 3.10.16 содержит перечень адресов приёмника для передач DMA 12 байт с использованием «полусловного» инкремента.

Таблица 3.10.16 — Цикл DMA для 12 байт с «полусловным» инкрементом

Начальные значения channel_cfg перед циклом DMA				
src_size=b00, dst_inc=b01, n_minus_1=b1011, cycle_ctrl=1, R_power=b11				
DMA передачи	Указатель конца данных	Счётчик	Отличие ¹⁾	Адрес
	0x5E7	11	0x16	0x5D1
	0x5E7	10	0x14	0x5D3
	0x5E7	9	0x12	0x5D5
	0x5E7	8	0x10	0x5D7
	0x5E7	7	0xE	0x5D9
	0x5E7	6	0xC	0x5DB
	0x5E7	5	0xA	0x5DD
	0x5E7	4	0x8	0x5DF
Значения channel_cfg после 2R передач DMA				
src_size=b00, dst_inc=b01, n_minus_1=b011, cycle_ctrl=1, R_power=b11				
	0x5E7	3	0x6	0x5E1
	0x5E7	2	0x4	0x5E3
	0x5E7	1	0x2	0x5E5
	0x5E7	0	0x0	0x5E7
Конечные значения channel_cfg после цикла DMA				
src_size=b00, dst_inc=b01, n_minus_1=0, cycle_ctrl=0 ²⁾ , R_power=b11				
¹⁾ Это значение, полученное после сдвига влево значения счётчика на количество разрядов, соответствующее dst_inc.				
²⁾ После окончания цикла DMA контроллер делает channel_cfg «неправильным», сбрасывая в 0 поле cycle_ctrl.				

Име. № подл.	Подп. и дата.
Взам. инв. №	Инв. № дубл.
Подп. и дата	
Име. № подл.	

Изм	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 109
-----	------	----------	-------	------	---	-------------

Основные положения регистровой модели контроллера:

— необходимо заполнять неиспользуемые или зарезервированные разряды регистров нулями при записи и игнорировать значения таких разрядов при считывании, кроме случаев, специально описанных в разделе;

— системный сброс или сброс по установке питания сбрасывает все регистры в нулевое состояние, кроме случаев, специально описанных в разделе;

— все регистры поддерживают доступ по чтению и записи, кроме случаев, специально описанных в настоящем разделе. Доступ по записи обновляет содержание регистра, а доступ по чтению возвращает содержимое регистра.

В таблице 3.10.17 представлен перечень регистров контроллера DMA.

Таблица 3.10.17 — Перечень регистров контроллера DMA

Смещение	Название регистра	Тип	Значение по сбросу	Описание
1	2	3	4	5
0x0	STATUS	RO	0x-0nn0000 ¹⁾	DMA -> STATUS Статусный регистр DMA
0x004	CFG	WO	–	DMA -> CFG Регистр конфигурации DMA
0x008	CTRL_BASE_PTR	RW	0x00000000	DMA -> CTRL_BASE_PTR. Регистр базового адреса управляющих данных каналов
0x00C	ALT_CTRL_BASE_PTR	RO	0x000000nn ²⁾	DMA -> ALT_CTRL_BASE_PTR. Регистр базового адреса альтернативных управляющих данных каналов
0x010	WAITONREQ_STATUS	RO	0x00000000	DMA -> WAITONREQ_STATUS. Регистр статуса ожидания запроса на обработку каналов
0x014	CHNL_SW_REQUEST	WO	–	DMA -> CHNL_SW_REQUEST. Регистр программного запроса на обработку каналов
0x018	CHNL_USEBURST_SET	RW	0x00000000	DMA -> CHNL_USEBURST_SET. Регистр установки пакетного обмена каналов

Изм.	Лист	№ докум.	Подп.	Дата	Смещение	Название регистра	Тип	Значение по сбросу	Описание
					1	2	3	4	5
					0x0	STATUS	RO	0x-0nn0000 ¹⁾	DMA -> STATUS Статусный регистр DMA
					0x004	CFG	WO	–	DMA -> CFG Регистр конфигурации DMA
					0x008	CTRL_BASE_PTR	RW	0x00000000	DMA -> CTRL_BASE_PTR. Регистр базового адреса управляющих данных каналов
					0x00C	ALT_CTRL_BASE_PTR	RO	0x000000nn ²⁾	DMA -> ALT_CTRL_BASE_PTR. Регистр базового адреса альтернативных управляющих данных каналов
					0x010	WAITONREQ_STATUS	RO	0x00000000	DMA -> WAITONREQ_STATUS. Регистр статуса ожидания запроса на обработку каналов
					0x014	CHNL_SW_REQUEST	WO	–	DMA -> CHNL_SW_REQUEST. Регистр программного запроса на обработку каналов
					0x018	CHNL_USEBURST_SET	RW	0x00000000	DMA -> CHNL_USEBURST_SET. Регистр установки пакетного обмена каналов
					Микросхема интегральная 5023BC016				Лист
					Техническое описание				
					Версия 2.04 от 01.09.2017 г.				110

Продолжение таблицы 3.10.17

1	2	3	4	5
0x01C	CHNL_USEBURST_CLR	WO	—	DMA -> CHNL_USEBURST_CLR Регистр сброса пакетного обмена каналов
0x020	CHNL_REQ_MASK_SET	RW	0x00000000	DMA -> CHNL_REQ_MASK_SET Регистр маскирования запросов на обслуживание каналов
0x024	CHNL_REQ_MASK_CLR	WO	—	DMA -> CHNL_REQ_MASK_CLR Регистр очистки маскирования запросов на обслуживание каналов
0x028	CHNL_ENABLE_SET	RW	0x00000000	DMA -> CHNL_ENABLE_SET. Регистр установки разрешения каналов
0x02C	CHNL_ENABLE_CLR	WO	—	DMA -> CHNL_ENABLE_CLR. Регистр сброса разрешения каналов
0x030	CHNL_PRI_ALT_SET	RW	0x00000000	DMA -> CHNL_PRI_ALT_SET. Регистр установки первичной/альтернативной структуры управляющих данных каналов
0x034	CHNL_PRI_ALT_CLR	WO	—	DMA -> CHNL_PRI_ALT_CLR. Регистр сброса первичной/альтернативной структуры управляющих данных каналов
0x038	CHNL_PRIORITY_SET	RW	0x00000000	DMA -> CHNL_PRIORITY_SET. установки приоритета каналов
0x03C	CHNL_PRIORITY_CLR	WO	—	DMA -> CHNL_PRIORITY_CLR. Регистр сброса приоритета каналов
0x040-0x048	—	—	—	Зарезервировано
0x04C	ERR_CLR	RW	0x00000000	DMA -> ERR_CLR. Регистр сброса флага ошибки
0x050-0xDFC	—	—	—	Зарезервировано

¹⁾ Значение по сбросу зависит от количества каналов DMA, использованных в контроллере, а также от того, интегрирована ли схема тестирования.

²⁾ Значение по сбросу зависит от количества каналов DMA, использованных в контроллере.

3.10.3.7 Регистр STATUS (статусный регистр DMA)

Регистр STATUS имеет доступ только на чтение. При чтении регистр возвращает состояние контроллера DMA. Если контроллер находится в состоянии сброса, то чтение регистра запрещено.

Име. № подл.	Подп. и дата.
Взам. инв. №	Инв. № дубл.
Подп. и дата	

Изм	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 111
-----	------	----------	-------	------	---	-------------

Таблица 3.10.18 содержит формат данных, содержащихся в регистре STATUS.
В таблице 3.10.19 содержится назначение разрядов регистра dma_status.

Таблица 3.10.18 — Формат данных, содержащихся в регистре STATUS

№ бита	31 – 28	27 – 21	20 – 16	15 – 8	7 – 4	3 – 1	0
Доступ	RO	U	RO	U	RO	U	RO
Сброс	0	0	0	0	0	0	0
	test_status	–	chnls_minus1	–	state	–	master_enable

Таблица 3.10.19 — Описание регистра dma_status

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 28	test_status	Значение при чтении: 0x0 = контроллер не имеет интегрированной схемы тестирования; 0x1 = контроллер имеет интегрированную схему тестирования; 0x2 – 0xF = не определено
27 – 21	–	Не определено
20 – 16	chnls_minus1	Количество доступных каналов DMA минус 1. Например: b00000 = контроллер поддерживает 1 канал DMA; b00001 = контроллер поддерживает 2 канала DMA; b00010 = контроллер поддерживает 3 канала DMA; b11111 = контроллер имеет 32 канала DMA
15 – 8	–	Не определено
7 – 4	state	Текущее состояние автомата управления контроллера. Состояние может быть одним из следующих: b0000 = в покое; b0001 = чтение управляющих данных канала; b0010 = чтение указателя конца данных источника; b0011 = чтение указателя конца данных приёмника; b0100 = чтение данных источника; b0101 = запись данных в приёмник; b0110 = ожидание запроса на выполнение DMA; b0111 = запись управляющих данных канала; b1000 = приостановлен; b1001 = выполнен; b1010 = режим работы с периферией «Исполнение с изменением конфигурации»; b1011 – b1111 = не определено
3 – 1	–	Не определено
0	master_enable	Состояние контроллера: 0 = функционирование контроллера запрещено; 1 = функционирование контроллера разрешено.

3.10.3.8 Регистр CFG (регистр конфигурации DMA)

Регистр CFG имеет доступ только на запись. Регистр определяет состояние контроллера.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	Микросхема интегральная 5023BC016					Лист
					Техническое описание					
					Версия 2.04 от 01.09.2017 г.					112
					Изм.	Лист	№ докум.	Подп.	Дата	

Таблица 3.10.20 содержит формат данных регистра конфигурации DMA. В таблице 3.10.21 содержится назначение разрядов регистра dma_cfg.

Таблица 3.10.20 — Формат данных регистра CFG

№ бита	31 – 8	7 – 5	4 – 1	0
Доступ	U	WO	U	WO
Сброс	0	0	0	0
	–	chnl_prot_ctrl	–	master_enable

Таблица 3.10.21 — Описание регистра dma_cfg

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 8	–	Не определено, следует записывать 0.
7 – 5	chnl_prot_ctrl	Определяет уровни индикации сигналов HPROT[3:1] защиты шины АHB-Lite: Разряд 7 управляет сигналом HPROT[3] с целью индикации о появлении доступа с кэшированием; Разряд 6 управляет сигналом HPROT[2] с целью индикации о появлении доступа с буферизацией; Разряд 5 управляет сигналом HPROT[1] с целью индикации о появлении привилегированного доступа. Примечание: Если разряд[n] = 1, то соответствующий сигнал HPROT в состоянии «1»; Если разряд[n] = 0, то соответствующий сигнал HPROT в состоянии «0».
4 – 1	–	Не определено. Следует записывать 0.
0	master_enable	Определяет состояние контроллера: 0 — запрещает работу контроллера; 1 — разрешает работу контроллера.

3.10.3.9 Регистр CTRL_BASE_PTR (регистр базового адреса управляющих данных каналов)

Регистр CTRL_BASE_PTR имеет доступ на запись и чтение. Регистр определяет базовый адрес системной памяти размещения управляющих данных каналов.

Примечание — Контроллер не содержит внутреннюю память для хранения управляющих данных каналов.

Размер системной памяти, предназначенной контроллеру, зависит от количества каналов DMA, использующихся контроллером, а также от возможности использования альтернативных управляющих данных каналов. Поэтому количество

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.
Изм	Лист	№ докум.	Подп.	Дата
				Микросхема интегральная 5023BC016
				Техническое описание
				Версия 2.04 от 01.09.2017 г.
				Лист
				113

Если контроллер находится в состоянии сброса, то чтение регистра запрещено.
Таблица 3.10.22 содержит конфигурацию регистра `ctrl_base_ptr`.

№ бита	31 – 10	9 – 0
Доступ	RW	U
Сброс	0	0
	ctrl_base_ptr	–

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 10	ctrl_base_ptr	Указатель на базовый адрес первичной структуры управляющих данных.
9 – 0	–	Не определено. Следует записывать 0.

Регистр ALT_CTRL_BASE_PTR имеет доступ только на чтение. Регистр возвращает при чтении указатель базового адреса альтернативных управляющих данных каналов. Если контроллер находится в состоянии сброса, то чтение регистра запрещено. Этот регистр позволяет не производить вычисления базового адреса альтернативных управляющих данных каналов.

Таблица 3.10.24 содержит конфигурацию разрядов регистра базового адреса альтернативных управляющих данных каналов.

Таблица 3.10.24 — Конфигурация разрядов регистра базового адреса альтернативных управляющих данных каналов

№ бита	31 – 0
Доступ	RO
Сброс	0
	Alt_ctrl_base_ptr

Таблица 3.10.25 содержит назначение разрядов регистра alt_ctrl_base_ptr.

Таблица 3.10.25 — Описание регистра alt_ctrl_base_ptr

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 0	alt_ctrl_base_ptr	Указатель базового адреса альтернативной структуры управляющих данных

3.10.3.11 Регистр WAITONREQ_STATUS (регистр статуса ожидания запроса на обработку каналов)

Регистр WAITONREQ_STATUS имеет доступ только на чтение. Регистр возвращает при чтении состояние сигналов dma_waitonreq[]. Если контроллер находится в состоянии сброса, то чтение регистра запрещено.

Таблица 3.10.26 содержит конфигурацию регистра статуса ожидания запроса на обработку каналов.

Таблица 3.10.26 — Конфигурация разрядов регистра статуса ожидания запроса на обработку каналов

№ бита	23	2	1	0
Доступ	RO	RO	RO	RO
Сброс	0	0	0	0
	dma_waitonreg_status for dma_waitnreg[23]	...	dma_waitonreg_status for dma_waitnreg[2]	dma_waitonreg_status for dma_waitnreg[1]	dma_waitonreg_status for dma_waitnreg[0]

Таблица 3.10.27 содержит конфигурацию разрядов регистра статуса ожидания запроса на обработку каналов.

Таблица 3.10.27 — Описание регистра dma_waitonreq_status

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
23 – 0	dma_waitonreq_status	Состояние сигналов ожидания запроса на обработку каналов DMA. При чтении: Разряд [C] = 0 означает, что dma_waitonreq[C] в состоянии «0» Разряд [C] = 1 означает, что dma_waitonreq[C] в состоянии «1»

Име. № подл.	Подп. и дата	Взам. име. №	Име. № дубл.	Подп. и дата.
Изм	Лист	№ докум.	Подп.	Дата
Микросхема интегральная 5023BC016				Лист
Техническое описание				
Версия 2.04 от 01.09.2017 г.				115

3.10.3.12 Регистр CHNL_SW_REQUEST (регистр программного запроса на обработку каналов)

Данный регистр имеет доступ только на запись. Регистр позволяет устанавливать программно запрос на выполнение цикла DMA.

Таблица 3.10.28 содержит конфигурацию разрядов регистра программного запроса на обработку каналов.

Таблица 3.10.28 — Регистр программного запроса на обработку каналов

№ бита	31 – 24	23	2	1	0
Доступ	U	WO	WO	WO	WO
Сброс	0	0	0	0	0
		Cgnl_sw_request for channel [23]	...	Cgnl_sw_request for channel [2]	Cgnl_sw_request for channel [1]	Cgnl_sw_request for channel [0]

Таблица 3.10.29 содержит назначение разрядов регистра chnl_sw_request.

Таблица 3.10.29 — Описание регистра chnl_sw_request

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
23 – 0	chnl_sw_request	Устанавливает соответствующий разряд для генерации программного запроса на выполнение цикла DMA по соответствующему каналу DMA. При записи: Разряд [C] = 0 означает, что запрос на выполнение цикла DMA по каналу «C» не будет установлен; Разряд [C] = 1 означает, что запрос на выполнение цикла DMA по каналу «C» будет установлен. Запись разряда, соответствующего нереализованному каналу, означает, что запрос на выполнение цикла DMA не будет установлен

3.10.3.13 Регистр CHNL_USEBURST_SET (регистр установки пакетного обмена каналов)

Регистр CHNL_USEBURST_SET имеет доступ на чтение и запись. Регистр отключает выполнение одиночных запросов по установке dma_sreq[], и поэтому

Ине. № подл.	Подп. и дата	Взам. ине. №	Ине. № дубл.	Подп. и дата.
Изм	Лист	№ докум.	Подп.	Дата
Микросхема интегральная 5023BC016				Лист
Техническое описание				116
Версия 2.04 от 01.09.2017 г.				

будут обрабатываться и исполняться только запросы по dma_req[]. Регистр возвращает при чтении состояние установок пакетного обмена.

Таблица 3.10.30 содержит конфигурацию разрядов регистра установки пакетного обмена каналов.

Таблица 3.10.30 — Конфигурация разрядов регистра установки пакетного обмена каналов

№ бита	31 – 24	23	2	1	0
Доступ	U	WO	WO	WO	WO
Сброс	0	0	0	0	0
		Chnl_useburst_set for channel [23]	...	Chnl_useburst_set for channel [2]	Chnl_useburst_set for channel [1]	Chnl_useburst_set for channel [0]

Таблица 3.10.31 содержит конфигурацию разрядов регистра chnl_useburst_set.

Таблица 3.10.31 — Описание регистра chnl_useburst_set

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
23 – 0	chnl_useburst_set	Отключает обработку запросов на выполнение циклов DMA от dma_sreq[] и возвращает при чтении состояние этих настроек. При чтении: Разряд [C] = 0 означает, что канал DMA C выполняет циклы DMA в ответ на запросы, полученные от dma_sreq[] и dma_req[]. Контроллер выполняет одиночные передачи или 2R передач. Разряд [C] = 1 означает, что канал DMA C выполняет циклы DMA в ответ на запросы, полученные только от dma_req[]. Контроллер выполняет 2R передач. При записи: Разряд [C] = 0 не даёт эффекта. Необходимо использовать chnl_useburst_clr регистр и установить соответствующий разряд C в 0; Разряд [C] = 1 отключает возможность обрабатывать запросы на выполнение циклов DMA, полученные от dma_sreq[]. Контроллер выполняет 2R передач. Запись разряда, соответствующего нереализованному каналу, не даёт эффекта.

После выполнения предпоследней передачи из 2R передач, в том случае, если число оставшихся передач (N) меньше чем 2R, контроллер сбрасывает разряд chnl_useburst_set в «0». Это позволяет выполнять оставшиеся передачи, используя dma_sreq[] и dma_req[].

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.

3.10.3.15 Регистр CHNL_REQ_MASK_SET (регистр маскирования запросов на обслуживание каналов)

Регистр CHNL_REQ_MASK_SET имеет доступ на чтение и запись. Регистр отключает установку запросов на выполнение циклов DMA на dma_sreq[] и dma_req[]. Регистр возвращает при чтении состояние установок маскирования запросов от dma_sreq[] и dma_req[] на обслуживание каналов.

Таблица 3.10.34 содержит конфигурацию разрядов регистра маскирования запросов на обслуживание каналов.

Таблица 3.10.34 — Конфигурация разрядов регистра маскирования запросов на обслуживание каналов

№ бита	31 – 24	23	2	1	0
Доступ	U	WO	WO	WO	WO
Сброс	0	0	0	0	0
		Chnl_reg_mask_set for dma_reg and dma_sreg [23]	...	Chnl_reg_mask_set for dma_reg and dma_sreg [2]	Chnl_reg_mask_set for dma_reg and dma_sreg [1]	Chnl_reg_mask_set for dma_reg and dma_sreg [0]

Таблица 3.10.35 содержит назначение разрядов регистра chnl_req_mask_set.

Таблица 3.10.35 — Описание регистра chnl_req_mask_set

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
23 – 0	chnl_req_mask_set	Отключает обработку запросов по dma_sreq[] и dma_req[] на выполнение циклов DMA от каналов и возвращает при чтении состояний этих настроек. При чтении: Разряд [C] = 0 означает, что канал DMA C выполняет циклы DMA в ответ на поступающие запросы; Разряд [C] = 1 означает, что канал DMA C не выполняет циклы DMA в ответ на поступающие запросы. При записи: Разряд [C] = 0 не дает эффекта. Необходимо использовать chnl_req_mask_clr регистр для разрешения установки запросов; Разряд [C] = 1 отключает установку запросов на выполнение циклов DMA, по dma_sreq[] и dma_req[]. Запись разряда, соответствующего нереализованному каналу, не даёт никакого эффекта.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата					
Изм	Лист	№ докум.	Подп.	Дата					
					Микросхема интегральная 5023BC016				
					Техническое описание				
					Версия 2.04 от 01.09.2017 г.				
					Лист				
					119				

3.10.3.16 Регистр CHNL_REQ_MASK_CLR (регистр очистки маскирования запросов на обслуживание каналов)

Регистр CHNL_REQ_MASK_CLR имеет доступ только на запись. Регистр разрешает установку запросов на выполнение циклов DMA на dma_sreq[] и dma_req[].

Таблица 3.10.36 содержит конфигурацию разрядов регистра очистки маскирования запросов на обслуживание каналов.

Таблица 3.10.36 — Конфигурация разрядов регистра очистки маскирования запросов на обслуживание каналов

№ бита	31 – 24	23	2	1	0
Доступ	U	WO	WO	WO	WO
Сброс	0	0	0	0	0
		Chnl_reg_mask_clr for dma_reg and dma_sreg [23]	...	Chnl_reg_mask_clr for dma_reg and dma_sreg [2]	Chnl_reg_mask_clr for dma_reg and dma_sreg [1]	Chnl_reg_mask_clr for dma_reg and dma_sreg [0]

Таблица 3.10.37 содержит назначение разрядов регистра chnl_req_mask_clr.

Таблица 3.10.37 — Описание регистра chnl_req_mask_clr

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
23 – 0	chnl_req_mask_clr	Установка соответствующего разряда разрешает установку запросов по dma_sreq[] и dma_req[] на выполнение циклов DMA от каналов. При записи: Разряд [C] = 0 не даёт эффекта. Необходимо использовать chnl_req_mask_set регистр для отключения установки запросов; Разряд [C] = 1 разрешает установку запросов на выполнение циклов DMA, по dma_sreq[] и dma_req[]. Запись разряда, соответствующего нереализованному каналу, не даёт никакого эффекта. Разряд [C] = 1 отключает установку запросов на выполнение циклов DMA, по dma_sreq[] и dma_req[]. Запись разряда, соответствующего нереализованному каналу, не даёт эффекта.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.
Изм	Лист	№ докум.	Подп.	Дата
Микросхема интегральная 5023BC016				Лист
Техническое описание				120
Версия 2.04 от 01.09.2017 г.				

3.10.3.17 Регистр CHNL_ENABLE_SET (регистр установки разрешения каналов)

Регистр CHNL_ENABLE_SET имеет доступ на чтение и запись. Регистр разрешает работу каналов DMA. Регистр возвращает при чтении состояние разрешений работы каналов DMA.

Таблица 3.10.38 содержит конфигурацию разрядов регистра установки разрешения каналов.

Таблица 3.10.38 — Конфигурация разрядов регистра установки разрешения каналов

№ бита	31 – 24	23	2	1	0
Доступ	U	WO	WO	WO	WO
Сброс	0	0	0	0	0
		Chnl_enable_set for cchannel [23]	...	Chnl_enable_set for cchannel [2]	Chnl_enable_set for cchannel [1]	Chnl_enable_set for cchannel [0]

Таблица 3.10.39 содержит назначение разрядов регистра chnl_enable_set.

Таблица 3.10.39 — Описание регистра chnl_enable_set

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
23 – 0	chnl_enable_set	Разрешает работу каналов DMA и возвращает при чтении состояние этих настроек. При чтении: Разряд [C] = 0 означает, что канал DMA C отключен; Разряд [C] = 1 означает, что функционирование канала DMA C разрешено. При записи: Разряд [C] = 0 не дает эффекта. Необходимо использовать chnl_enable_clr регистр для отключения канала; Разряд [C] = 1 разрешает работу канала DMA C. Запись разряда, соответствующего нереализованному каналу, не дает эффекта.

3.10.3.18 Регистр CHNL_ENABLE_CLR (регистр сброса разрешения каналов)

Регистр CHNL_ENABLE_CLR имеет доступ только на запись. Регистр запрещает функционирование каналов DMA.

Таблица 3.10.40 содержит конфигурацию разрядов регистра сброса разрешения каналов.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата					
Изм	Лист	№ докум.	Подп.	Дата					
					Микросхема интегральная 5023BC016				
					Техническое описание				
					Версия 2.04 от 01.09.2017 г.				
					Лист				
					121				

Таблица 3.10.40 — Конфигурация разрядов регистр сброса разрешения каналов

№ бита	31 – 24	23	2	1	0
Доступ	U	WO	WO	WO	WO
Сброс	0	0	0	0	0
		Chnl_enable_clr for cgannel [23]	...	Chnl_enable_clr for cgannel [2]	Chnl_enable_clr for cgannel [1]	Chnl_enable_clr for cgannel [0]

Таблица 3.10.41 содержит назначение разрядов регистра chnl_enable_clr.

Таблица 3.10.41 — Описание регистра chnl_enable_clr

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
23 – 0	chnl_enable_clr	Установка соответствующего разряда запрещает работу соответствующего канала DMA. При записи: Разряд [C] = 0 не дает эффекта. Необходимо использовать chnl_enable_set регистр для разрешения работы канала; Разряд [C] = 1 запрещает работу канала DMA C. Запись разряда, соответствующего нереализованному каналу, не дает никакого эффекта. Примечание: Контроллер может отключить канал DMA, установив соответствующий разряд в следующих случаях: — при завершении цикла DMA; — при чтении из channel_cfg с полем cycle_ctrl установленным vb000; — при появлении ошибки на шине AHB-Lite.

3.10.3.19 Регистр CHNL_PRI_ALT_SET (регистр установки первичной/альтернативной структуры управляющих данных каналов)

Регистр CHNL_PRI_ALT_SET имеет доступ на запись и чтение. Регистр разрешает работу канала DMA с использованием альтернативной структуры управляющих данных. Результатом чтения регистра является состояние каналов DMA (какую структуру управляющих данных использует каждый канал DMA).

Таблица 3.10.42 содержит конфигурацию разрядов регистра установки первичной/альтернативной структуры управляющих данных каналов.

Ине. № подл.	Подп. и дата	Взам. инв. №	Ине. № дубл.	Подп. и дата.

Таблица 3.10.42 — Конфигурация разрядов регистра установки первичной/альтернативной структуры управляющих данных каналов

№ бита	31 – 24	23	2	1	0
Доступ	U	WO	WO	WO	WO
Сброс	0	0	0	0	0
		Chnl_pri_alt_set for cgannel [23]	...	Chnl_pri_alt_set for cgannel [2]	Chnl_pri_alt_set for cgannel [1]	Chnl_pri_alt_set for cgannel [0]

Таблица 3.10.43 содержит назначение разрядов регистра chnl_pri_alt_set.

Таблица 3.10.43 — Описание регистра chnl_pri_alt_set

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
23 – 0	chnl_pri_alt_set	<p>Установка соответствующего разряда подключает использование альтернативных управляющих данных для соответствующего канала DMA, чтение возвращает состояние этих настроек.</p> <p>При чтении:</p> <p>Разряд [C] = 0 означает, что канал DMA C использует первичную структуру управляющих данных;</p> <p>Разряд [C] = 1 означает, что канал DMA C использует альтернативную структуру управляющих данных.</p> <p>При записи:</p> <p>Разряд [C] = 0 не дает эффекта. Необходимо использовать chnl_pri_alt_clr регистр для сброса разряда [C] в 0;</p> <p>Разряд [C] = 1 подключает использование альтернативной структуры управляющих данных каналом DMA C.</p> <p>Запись разряда, соответствующего нереализованному каналу, не даёт никакого эффекта.</p> <p>Примечание — Контроллер может переключить значение разряда chnl_pri_alt_set[C] в следующих случаях:</p> <ul style="list-style-type: none">— при завершении 4-х передач DMA указанных в первичной структуре управляющих данных при выполнении цикла DMA в режимах работы с памятью или периферией «исполнение с изменением конфигурации»;— при завершении всех передач DMA указанных в первичной структуре управляющих данных при выполнении цикла DMA в режиме «Пинг-понг»;— при завершении всех передач DMA указанных в альтернативной структуре управляющих данных при выполнении цикла DMA в режимах:— «пинг-понг»;— работа с памятью в режиме scatter-gather;— работа с периферией в режиме scatter-gather.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.

3.10.3.20 Регистр CHNL_PRI_ALT_CLR (регистр сброса первичной/альтернативной структуры управляющих данных каналов)

Регистр CHNL_PRI_ALT_CLR имеет доступ только на запись. Регистр разрешает работу канала DMA с использованием первичной структуры управляющих данных.

Таблица 3.10.44 содержит конфигурацию разрядов регистра сброса первичной/альтернативной структуры управляющих данных каналов.

Таблица 3.10.44 — Конфигурация разрядов регистра сброса первичной/альтернативной структуры управляющих данных каналов

№ бита	31 – 24	23	2	1	0
Доступ	U	WO	WO	WO	WO
Сброс	0	0	0	0	0
		Chnl_pri_alt_clr for cchannel [23]	...	Chnl_pri_alt_clr for cchannel [2]	Chnl_pri_alt_clr for cchannel [1]	Chnl_pri_alt_clr for cchannel [0]

Таблица 3.10.45 содержит назначение разрядов регистра chnl_pri_alt_clr.

Таблица 3.10.45— Описание регистра chnl_pri_alt_clr

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
23 – 0	chnl_pri_alt_clr	<p>Установка соответствующего разряда подключает использование первичных управляющих данных для соответствующего канала DMA. При записи: «Разряд [C] = 0» не даёт эффекта. Необходимо использовать chnl_pri_alt_set регистр для выбора альтернативных управляющих данных; «Разряд [C] = 1» подключает использование первичной структуры управляющих данных каналом DMA C. Запись разряда, соответствующего нереализованному каналу, не даёт эффекта.</p> <p>Примечание — Контроллер может изменять значение разряда chnl_pri_alt_clr[C] в случаях завершения следующих передач:</p> <ul style="list-style-type: none"> — 4-х передач DMA указанных в первичной структуре управляющих данных при выполнении цикла DMA в режиме «Scatter-gather»; — всех передач DMA указанных в первичной структуре управляющих данных при выполнении цикла DMA в режиме «пинг-понг»; — всех передач DMA указанных в альтернативной структуре управляющих данных при выполнении цикла DMA в режимах: «пинг-понг»; «Scatter-gather» с памятью; «Scatter-gather» с периферией

Име. № подл.	Подп. и дата	Взам. име. №	Име. № дубл.	Подп. и дата.
Изм	Лист	№ докум.	Подп.	Дата
Микросхема интегральная 5023BC016				Лист
Техническое описание				124
Версия 2.04 от 01.09.2017 г.				

3.10.3.21 Регистр CHNL_PRIORITY_SET (регистр установки приоритета каналов)

Регистр CHNL_PRIORITY_SET имеет доступ на запись и чтение. Регистр позволяет присвоить высокий приоритет каналу DMA. Чтение регистра возвращает состояние приоритета каналов DMA.

Таблица 3.10.46 содержит конфигурацию разрядов регистра установки приоритета каналов.

Таблица 3.10.46 — Конфигурация разрядов регистра установки приоритета каналов

№ бита	31 – 24	23	2	1	0
Доступ	U	WO	WO	WO	WO
Сброс	0	0	0	0	0
		Chnl_prioriti_set for cchannel [23]	...	Chnl_prioriti_set for cchannel [2]	Chnl_prioriti_set for cchannel [1]	Chnl_prioriti_set for cchannel [0]

Таблица 3.10.47 содержит назначение разрядов регистра chnl_priority_set.

Таблица 3.10.47 — Описание регистра chnl_priority_set

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
23 – 0	chnl_priority_set	<p>Установка высокого приоритета каналу DMA, чтение возвращает состояние приоритета каналов DMA.</p> <p>При чтении:</p> <p>Разряд [C] = 0 означает, что каналу DMA «С» присвоен уровень приоритета по умолчанию;</p> <p>Разряд [C] = 1 означает, что каналу DMA «С» присвоен высокий уровень приоритета.</p> <p>При записи:</p> <p>Разряд [C] = 0 не дает эффекта. Необходимо использовать chnl_priority_clr регистр для установки каналу С уровня приоритета по умолчанию;</p> <p>Разряд [C] = 1 устанавливает каналу DMA «С» высокий уровень приоритета.</p> <p>Запись разряда, соответствующего нереализованному каналу, не дает эффекта.</p>

3.10.3.22 Регистр CHNL_PRIORITY_CLR (регистр сброса приоритета каналов)

Регистр CHNL_PRIORITY_CLR имеет доступ только на запись. Регистр позволяет присвоить каналу DMA-уровень приоритета по умолчанию.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.
Изм	Лист	№ докум.	Подп.	Дата
Микросхема интегральная 5023BC016				Лист
Техническое описание				125
Версия 2.04 от 01.09.2017 г.				

Таблица 3.10.48 содержит конфигурацию разрядов регистра сброса приоритета каналов.

Таблица 3.10.48 — Конфигурация разрядов регистра сброса приоритета каналов

№ бита	31 – 24	23	2	1	0
Доступ	U	WO	WO	WO	WO
Сброс	0	0	0	0	0
		Chnl_prioriti_clr for cchannel [23]	...	Chnl_prioriti_clr for cchannel [2]	Chnl_prioriti_clr for cchannel [1]	Chnl_prioriti_clr for cchannel [0]

Таблица 3.10.49 содержит назначение разрядов регистра chnl_priority_clr.

Таблица 3.10.49 — Описание регистра chnl_priority_clr

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
23 – 0	chnl_priority_clr	Установка разряда присваивает соответствующему каналу DMA уровень приоритета по умолчанию. При записи: Разряд [C] = 0 не даёт эффекта. Необходимо использовать chnl_priority_set регистр для установки каналу «С» высокого уровня приоритета. Разряд [C] = 1 устанавливает каналу DMA «С» уровень приоритета по умолчанию. Запись разряда, соответствующего нереализованному каналу, не дает эффекта.

3.10.3.23 Регистр ERR_CLR (регистр сброса флага ошибки)

Регистр ERR_CLR имеет доступ на запись и чтение. Регистр позволяет сбрасывать сигнал dma_err в «0». Чтение регистра возвращает состояние сигнала dma_err.

Таблица 3.10.50 содержит конфигурацию разрядов регистра сброса флага ошибки.

Таблица 3.10.50 — Конфигурация разрядов регистра сброса флага ошибки

№ бита	31 – 1	0
Доступ	U	RW
Сброс	0	0
	–	err_clr

Ине. № подл.	Подп. и дата	Взам. инв. №	Ине. № дубл.	Подп. и дата.

Изм	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 126
-----	------	----------	-------	------	---	-------------

Таблица 3.10.51 содержит назначение разрядов регистра err_clr.

Таблица 3.10.51 — Описание регистра err_clr

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 1	–	Не определено. Следует записывать 0
23 – 0	chnl_priority_clr	<p>Установка сигнала в состояние 0, чтение возвращает состояние сигнала (флага) dma_err.</p> <p>При чтении:</p> <p>Разряд [C] = 0 означает, что dma_err находится в состоянии «0»;</p> <p>Разряд [C] = 1 означает, что dma_err находится в состоянии «1».</p> <p>При записи:</p> <p>Разряд [C] = 0 не дает эффекта. Состояние dma_err останется неизменным;</p> <p>Разряд [C] = 1 сбрасывает сигнал (флаг) dma_err в состояние «0».</p> <p>Для целей тестирования возможно использовать регистр err_set, чтобы установить сигнал dma_err в состояние «1».</p> <p>Примечание:</p> <p>При сбросе сигнала dma_err одновременно с появлением ошибки на шине АНВ-Lite, то приоритет отдается ошибке и, следовательно, значение регистра (и dma_err) останется неизменным (несброшенным).</p>

3.11 Контроллер интерфейса SPI

3.11.1 Общие положения

Контроллер интерфейса SPI выполняет функции интерфейса последовательной синхронной связи в режиме ведущего и ведомого устройства и обеспечивает обмен данными с подключенным ведомым или ведущим периферийным устройством в соответствии с одним из протоколов:

- интерфейс SPI фирмы Motorola;
- интерфейс SSI фирмы Texas Instruments;
- интерфейс Microwire фирмы National Semiconductor.

Как в ведущем, так и в ведомом режиме работы модуль SPI обеспечивает:

- преобразование данных, размещенных во внутреннем буфере FIFO передатчика (восемь 16-разрядных ячеек данных), из параллельного в последовательный формат;
- преобразование данных из последовательного в параллельный формат и их запись в аналогичный буфер FIFO приёмника (восемь 16-разрядных ячеек данных).

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.
Изм	Лист	№ докум.	Подп.	Дата
Микросхема интегральная 5023BC016				Лист
Техническое описание				127
Версия 2.04 от 01.09.2017 г.				

Модуль SPI формирует сигналы прерываний по следующим событиям:

- необходимость обслуживания буферов FIFO приёмника и передатчика;
- переполнение буфера FIFO приёмника;
- наличие данных в буфере FIFO приёмника по истечении времени таймаута.

3.11.2 Основные характеристики интерфейса SPI

Интерфейс SPI имеет следующие характеристики:

- функционирование в ведущем и в ведомом режимах;
- программное управление скоростью обмена;
- программный выбор одного из интерфейсов обмена: SPI, Microwire, SSI;
- программируемая длительность информационного кадра от 4 до 16 бит;
- независимое маскирование прерываний от буфера FIFO передатчика, буфера FIFO приёмника, а также по переполнению буфера приёмника;
- доступна возможность тестирования по шлейфу, соединяющему вход с выходом;
- поддержка прямого доступа к памяти (DMA).

Структурная схема модуля SPI представлена на рисунке 3.11.1.

Состав модуля: независимые буферы приёма и передачи (8 ячеек по 16 бит) с организацией доступа типа FIFO.

3.11.3 Программируемые параметры интерфейса SPI

Следующие ключевые параметры интерфейса SPI могут быть заданы программно:

- режим функционирования периферийного устройства — ведущее или ведомое;
- разрешение или запрет функционирования;
- формат информационного кадра;
- скорость передачи данных;
- фаза и полярность тактового сигнала;
- размер блока данных — от 4 до 16 бит;
- маскирование прерываний.

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>					Лист
Изм.	Лист	№ докум.	Подп.	Дата						128

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.

— программное задание фазы и полярности тактового сигнала.

Интерфейс Microwire фирмы National Semiconductors обеспечивает полудуплексный обмен данными с использованием восьмибитных управляющих последовательностей.

— полнодуплексный обмен данными по четырехпроводной линии;

— возможность перевода линии передачи данных в третье (высокоимпедансное) состояние.

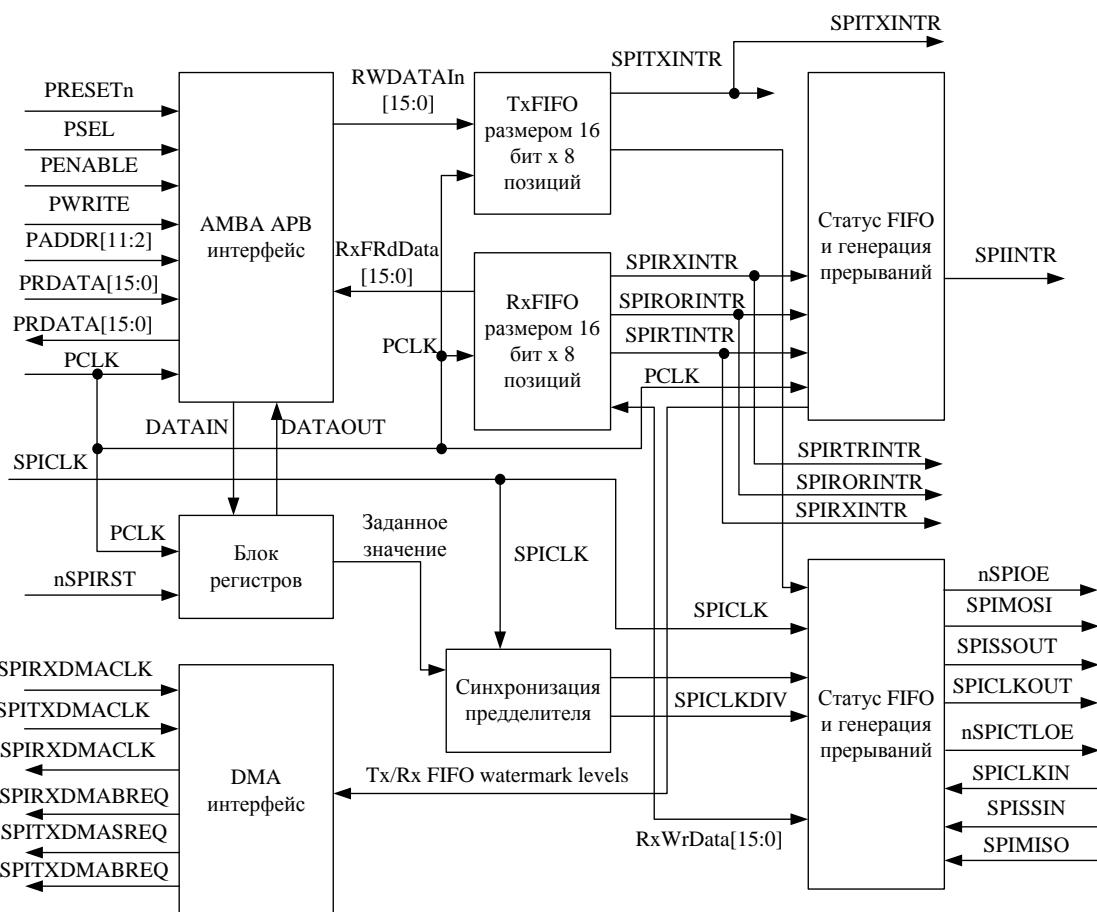


Рисунок 3.11.1 — Структурная схема модуля SPI

3.11.5 Функциональное описание интерфейса SPI

3.11.5.1 Общие сведения об интерфейсе SPI

Модуль SPI содержит программируемые делители частоты, формирующие тактовый сигнал обмена данными SPI_SCK. Скорость передачи данных может достигать более 2 МГц, в зависимости от частоты SPICLK и характеристик подключенного периферийного устройства.

Режим обмена данными, формат информационного кадра и количество бит данных задаются программно с помощью регистров управления CR0 и CR1.

Модуль формирует четыре независимо маскируемых прерывания:

- SPITXINTR — запрос на обслуживание буфера передатчика;
- SPIRXINTR — запрос на обслуживание буфера приёмника;
- SPIRORINTR — переполнение приёмного буфера FIFO;
- SPIRTINTR — таймаут ожидания чтения данных из приёмного FIFO.

Кроме того, формируется общий сигнал прерывания SPIINTR, возникающий в случае активности одного из вышеуказанных независимых немаскированных прерываний, который идёт на контроллер NVIC.

Модуль формирует сигналы запроса на прямой доступ к памяти (DMA) для совместной работы с контроллером DMA.

В зависимости от режима работы модуля сигнал SPISSOUT используется либо для кадровой синхронизации (интерфейс SSI, активное состояние — высокий уровень), либо для выбора ведомого режима (интерфейсы SPI и Microwire, активное состояние — низкий уровень).

3.11.5.2 Блок формирования тактового сигнала

В режиме ведущего устройства модуль формирует тактовый сигнал обмена данными SPI_SCK с помощью внутреннего делителя частоты, состоящего из двух последовательно соединённых счётчиков без цепи сброса.

Путём записи значения в регистр SPICPSR можно задать коэффициент предварительного деления частоты в диапазоне от 2 до 254 с шагом 2. Так как младший значащий разряд коэффициента деления не используется, то исключается

Име. № подл.	Подп. и дата		Име. № дубл.		Подп. и дата						
	Взам. име. №										
						Лист					
Изм	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.		130				

прерываний, который идёт на контроллер NVIC.

Модуль формирует сигналы запроса на прямой доступ к памяти (DMA) для совместной работы с контроллером DMA.

В зависимости от режима работы модуля сигнал SPISSOUT используется либо для кадровой синхронизации (интерфейс SSI, активное состояние — высокий уровень), либо для выбора ведомого режима (интерфейсы SPI и Microwire, активное состояние — низкий уровень).

3.11.5.2 Блок формирования тактового сигнала

В режиме ведущего устройства модуль формирует тактовый сигнал обмена данными SPI_SCK с помощью внутреннего делителя частоты, состоящего из двух последовательно соединённых счётчиков без цепи сброса.

Путём записи значения в регистр SPICPSR можно задать коэффициент предварительного деления частоты в диапазоне от 2 до 254 с шагом 2. Так как младший значащий разряд коэффициента деления не используется, то исключается

возможность деления частоты на нечетный коэффициент деления. Это, в свою очередь, гарантирует формирование тактового сигнала симметричной формы (с одинаковой длительностью полупериодов высокого и низкого уровней).

Сформированный описанным образом сигнал далее поступает на второй делитель частоты, с выхода которого и снимается тактовый сигнал обмена данными SPI_SCK.

Коэффициент деления второго делителя задается программно в диапазоне от 1 до 256, путем записи соответствующего значения в регистр управления SPICR0.

3.11.5.3 Буфер FIFO передатчика

Буфер передатчика имеет ширину 16 бит, глубину 8 слов, схему организации доступа типа FIFO. Данные от центрального процессора сохраняются в буфере до тех пор, пока не будут считаны блоком передачи данных.

3.11.5.4 Буфер FIFO приёмника

Буфер приёмника имеет ширину 16 бит, глубину 8 слов, схему организации доступа типа FIFO. Принятые от периферийного устройства данные сохраняются в этом буфере блоком приёма данных до тех пор, пока не будут считаны центральным процессором.

3.11.5.5 Блок приёма и передачи данных

В режиме ведущего устройства модуль формирует тактовый сигнал обмена данными SPI_SCK для подключенных ведомых устройств. Как было описано ранее, данный сигнал формируется путём деления частоты сигнала SPICLK.

Блок передатчика последовательно считывает данные из буфера FIFO передатчика и производит их преобразование из параллельной формы в последовательную. Далее поток последовательных данных и элементов кадровой синхронизации, тактированный сигналом SPI_SCK, передаётся по линии SPI_MOSI к подключенным ведомым устройствам.

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>					Лист
Изм.	Лист	№ докум.	Подп.	Дата						131

Блок приёмника выполняет преобразование данных, поступающих синхронно с линии SPI_MISO, из последовательной в параллельную форму, после чего загружает их в буфер FIFO приёмника, откуда они могут быть считаны процессором.

В режиме ведомого устройства тактовый сигнал обмена данными формируется одним из подключенных к модулю периферийных устройств и поступает по линии SPI_SCK.

При этом блок передатчика, тактируемый этим внешним сигналом, считывает данные из буфера FIFO, преобразует их из параллельной формы в последовательную, после чего выдает поток последовательных данных и элементов кадровой синхронизации в линию SPI_MOSI.

Аналогично, блок приёмника выполняет преобразование данных, поступающих с линии SPI_MISO синхронно с сигналом SPI_SCK, из последовательной в параллельную форму, после чего загружает их в буфер FIFO приёмника, откуда последние могут быть считаны процессором.

3.11.5.6 Блок формирования прерываний

Модуль SPI генерирует независимые маскируемые прерывания с активным высоким уровнем. Кроме того, формируется комбинированное прерывание путём объединения указанных независимых прерываний по схеме ИЛИ.

Комбинированный сигнал прерывания подаётся на контроллер прерываний NVIC, при этом появляется дополнительная возможность маскирования устройства в целом, что облегчает построение модульных драйверов устройств.

3.11.5.7 Интерфейс прямого доступа к памяти

Модуль обеспечивает интерфейс с контроллером DMA согласно схеме взаимодействия приёмопередатчика и контроллера DMA.

3.11.6 Функционирование модуля SPI

3.11.6.1 Конфигурирование приёмопередатчика

После сброса функционирование блоков приёмопередатчика запрещается до выполнения процедуры задания конфигурации.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>					Лист
Изм.	Лист	№ докум.	Подп.	Дата						132

Для этого необходимо выбрать ведущий или ведомый режим работы устройства, а также используемый протокол передачи данных (SPI фирмы Motorola, SSI фирмы Texas Instruments, либо Microwire фирмы National Semiconductor), после чего записать необходимую информацию в регистры управления CR0 и CR1.

Кроме того, для установки требуемой скорости передачи данных необходимо выбрать параметры блока формирования тактового сигнала с учётом значения частоты сигнала SPICLK и записать соответствующую информацию в регистр PSR.

3.11.6.2 Разрешение работы приёмопередатчика

Разрешение осуществляется путём установки бита SSE регистра управления CR1. Буфер FIFO передатчика может быть либо проинициализирован путём записи в него до восьми 16-разрядных слов заблаговременно перед установкой этого бита, либо может заполняться передаваемыми данными в процедуре обслуживания прерывания.

После разрешения работы модуля приёмопередатчик начинает обмен данными по линиям SPI_MOSI и SPI_MISO.

3.11.6.3 Соотношения между тактовыми сигналами

В модуле имеется ограничение на соотношение между частотами тактовых сигналов CPU_CLK и SPICLK. Частота SPICLK должна быть меньше или равна частоте CPU_CLK

$$F_{SPICLK} \leq F_{PCLK}. \tag{2}$$

Выполнение этого требования гарантирует синхронизацию сигналов управления, передаваемых из зоны действия тактового сигнала SPICLK в зону действия сигнала CPU_CLK в течение времени, меньшего продолжительности передачи одного информационного кадра.

В режиме ведомого устройства сигнал SPI_SCK от ведущего внешнего устройства поступает на схемы синхронизации, задержки и обнаружения фронта. Сигнал SPI_MOSI имеет меньшее время установки по отношению к заднему фронту SPI_SCK, по которому и происходит считывание данных из линии. Время установки

Ине. № подл.	Подп. и дата
Взам. инв. №	Инв. № дубл.
Подп. и дата	
Ине. № подл.	Подп. и дата

					Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 133
Изм	Лист	№ докум.	Подп.	Дата		

и удержания сигнала SPI_MISO по отношению к сигналу SPI_SCK должно выбираться с запасом, гарантирующим правильное считывание данных.

Выбор частоты тактового сигнала SPICLK должен обеспечивать поддержку требуемого диапазона скоростей обмена данными. Отношение минимальной частоты сигнала SPICLK к максимальной частоте сигнала SPI_SCK в режиме ведомого устройства равно 12, в режиме ведущего — двум.

Так, в режиме ведущего устройства для обеспечения максимальной скорости обмена 1,8432 Мбит/с частота сигнала SPICLK должна составлять не менее 3,6864 МГц. В этом случае в регистр CPSR должно быть записано значение 2, а поле SCR[7:0] регистра CR0 должно быть установлено в «0».

В режиме ведомого устройства для обеспечения той же информационной скорости необходимо использовать тактовый сигнал SPICLK с частотой не менее 22,12 МГц. При этом в регистр CPSR должно быть записано значение 12, а поле SCR[7:0] регистра CR0 должно быть установлено в «0».

3.11.6.4 Программирование регистра управления CR0

Регистр CR0 выполняет следующие функции:

- установку скорости информационного обмена;
- выбора одного из трёх протоколов обмена данными;
- выбора размера слова данных.

Скорость информационного обмена зависит от частоты внешнего тактового сигнала и коэффициента деления блока формирования тактового сигнала. Последний задаётся совместно значением поля SerialClockRate (SCR) регистра SPICR и значением поля CPSDVSR (clockprescaledivisorvalue – коэффициент деления тактового сигнала) регистра SPICPSR.

Формат информационного кадра задаётся путём установки значения поля FRF, а размер слова данных — путём установки значения поля DSS регистра SPICR0.

Для протокола SPI фирмы Motorola также задаются полярность и фаза сигнала (биты SPH и SPO).

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	Микросхема интегральная 5023BC016					Лист
					Техническое описание					
					Версия 2.04 от 01.09.2017 г.					134
					Изм	Лист	№ докум.	Подп.	Дата	

3.11.6.5 Программирование регистра управления CR1

Регистр SPICR1 выполняет следующие функции:

- выбор ведущего или ведомого режима функционирования приёмопередатчика;
- включение режима проверки канала по шлейфу;
- разрешение или запрещение работы модуля.

Выбор ведущего режима осуществляется путём записи «0» в поле MS регистра SPICR1 (это значение устанавливается после сброса автоматически).

Запись «1» в поле MS переводит приёмопередатчик в режим ведомого устройства. В этом режиме разрешение или запрещение формирования сигнала передатчика SPI_MOSI осуществляется путём установки бита SOD (Slave mode Output Disable — запрет линии SPI_MOSI для ведомого режима) регистра CR1. Указанная функция полезна при подключении к одной линии нескольких подчинённых устройств.

Для того чтобы разрешить функционирование приёмопередатчика, необходимо установить в 1 бит Synchronous Serial Port Enable (SSE).

3.11.6.6 Формирование тактового сигнала обмена данными

Тактовый сигнал обмена данными формируется путём деления частоты тактового сигнала SPICLK. На первом этапе формирования частота этого сигнала делится на четный коэффициент CPSDVSR, лежащий в диапазоне от 2 до 254, доступный для программирования через регистр CPSR. Сформированный сигнал далее поступает на делитель частоты с коэффициентом $(1 + SCR)$ от 1 до 256, где значение SCR доступно для программирования через CR0.

Частота выходного тактового сигнала обмена данными SPI_SCK определяется следующим соотношением

$$F_{SPI_SCKOUT} = F_{SPICLK} / (CPSDVR \cdot (1 + SCR)). \quad (3)$$

Например, в случае, если частота сигнала SPICLK составляет 3,6864 МГц, а значение CPSDVSR = 2, частота сигнала SPI_SCK лежит в интервале от 7,2 кГц до 1,8432 МГц.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	<div>Микросхема интегральная 5023BC016</div> <div>Техническое описание</div> <div>Версия 2.04 от 01.09.2017 г.</div>					Лист
Изм.	Лист	№ докум.	Подп.	Дата						135

3.11.6.7 Формат информационного кадра

Каждый информационный кадр содержит в зависимости от запрограммированного значения от 4 до 16 бит данных. Передача данных начинается со старшего значащего разряда. Возможно выбрать три базовых структуры построения кадра:

- SSI фирмы Texas Instruments;
- SPI фирмы Motorola;
- Microwire фирмы National Semiconductor.

Во всех трёх режимах построения кадра тактовый сигнал SPI_SCK формируется только тогда, когда приёмопередатчик готов к обмену данными. Перевод сигнала SPI_SCK в неактивное состояние используется как признак таймаута приёмника, то есть наличия в буфере приёмника необработанных данных по истечении заданного интервала времени.

В режимах SPI и Microwire выходной сигнал кадровой синхронизации передатчика SPI_SS имеет активный низкий уровень и поддерживается в низком уровне в течение всего периода передачи информационного кадра.

В режиме построения кадра SSI перед началом каждого информационного кадра на выходе SPI_SS формируется импульс с длительностью, равной одному тактовому интервалу обмена данными. В этом режиме приёмопередатчик SPI, равно как и ведомые периферийные устройства, передаёт данные в линию по переднему фронту сигнала SPI_SCK, а считывает данные из линии по заднему фронту этого сигнала.

В отличие от полнодуплексных режимов передачи данных SSI и SPI, режим Microwire фирмы National Semiconductor использует специальный способ обмена данными между ведущим и ведомым устройством, функционирующий в режиме полудуплекса. В указанном режиме на внешнее ведомое устройство перед началом передачи информационного кадра посылается специальная восьмибитная управляющая последовательность. В течение всего времени передачи этой последовательности приёмник не обрабатывает каких-либо входных данных. После того, как сигнал передан и декодирован ведомым устройством, оно выдерживает

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	<div>Микросхема интегральная 5023BC016</div> <div>Техническое описание</div> <div>Версия 2.04 от 01.09.2017 г.</div>					Лист
Изм.	Лист	№ докум.	Подп.	Дата						136

паузу в один тактовый интервал после передачи последнего бита управляющей последовательности, после чего передает в адрес ведущего устройства запрошенные данные. Длительность блока данных от ведомого устройства может составлять от 4 до 16 бит, таким образом общая длительность информационного кадра составляет от 13 до 25 бит.

3.11.6.8 Формат синхронного обмена SSI фирмы Texas Instruments

Временные диаграммы, отображающие формат синхронного обмена протокола SSI (единичный обмен) представлены на рисунке 3.11.2.

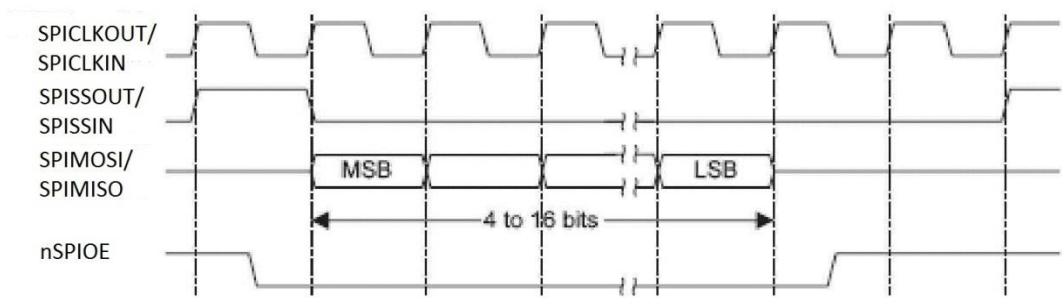


Рисунок 3.11.2 — Временные диаграммы, отображающие формат синхронного обмена протокола SSI (единичный обмен)

В режиме синхронного обмена SSI при неактивном приёмопередатчике SPI сигналы SPI_SCK и SPI_SS переводятся в низкий логический уровень, а линия передачи данных SPI_MOSI поддерживается в третьем состоянии.

После появления хотя бы одного элемента в буфере FIFO передатчика сигнал SPI_SS переводится в высокий логический уровень на время, соответствующее одному периоду сигнала SPI_SCK. Значение из буфера FIFO, при этом переносится в сдвиговый регистр блока передатчика. По следующему переднему фронту сигнала SPI_SCK старший значащий разряд информационного кадра (4 – 16 бит данных) выдаётся на выход линии SPI_MOSI и т.д.

В режиме приёма данных как модуль SPI, так и ведомое внешнее устройство последовательно загружают биты данных в сдвиговый регистр по заднему фронту сигнала SPI_SCK. Принятые данные переносятся из сдвигового регистра в буфер FIFO после загрузки в него младшего значащего бита данных по очередному переднему фронту сигнала SPI_SCK.

Име. № подл.	Подп. и дата	Взам. инв. №	Име. № дубл.	Подп. и дата.

Временные диаграммы последовательного синхронного обмена по протоколу SSI фирмы Texas Instruments представлены на рисунке 3.11.3.

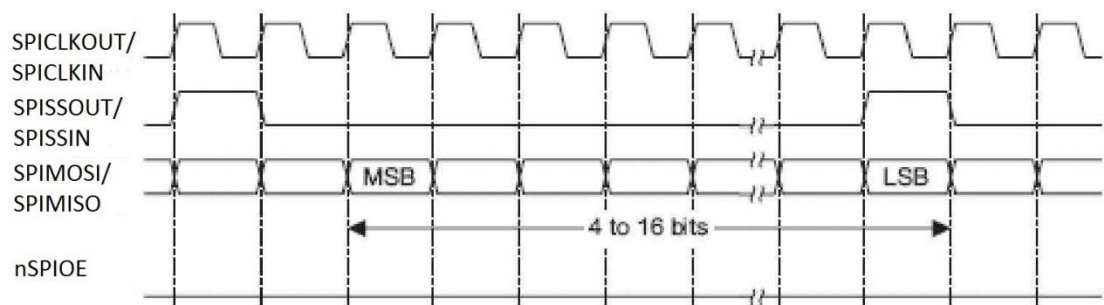


Рисунок 3.11.3 — Формат синхронного обмена протокола SSI (непрерывный обмен)

3.11.6.9 Формат синхронного обмена SPI фирмы Motorola

Интерфейс SPI фирмы Motorola организован посредством четырех сигнальных линий, при этом сигнал SPI_SS выполняет функцию выбора ведомого устройства. Главной особенностью протокола SPI является возможность выбора состояния и фазы сигнала SPI_SCK в режиме ожидания (неактивном приёмопередатчике) путём задания значений бит SPO и SPH регистра управления SPISCR0.

3.11.6.10 Выбор полярности тактового сигнала

Для выбора полярности тактового сигнала используется бит SPO.

Если бит SPO равен «0», то в режиме ожидания линия SPI_SCK переводится в низкий логический уровень. В противном случае при отсутствии обмена данными линия SPI_SCK переводится в высокий логический уровень.

3.11.6.11 Выбор фазы тактового сигнала

Для выбора фазы тактового сигнала используется бит SPH.

Значение бита SPH определяет фронт тактового сигнала, по которому осуществляется выборка данных и изменение состояния на выходе линии.

В случае если бит SPH установлен в «0», регистрация данных приёмником осуществляется после первого обнаружения фронта тактового сигнала, в противном случае — после второго.

3.11.6.12 Формат синхронного обмена SPI фирмы Motorola, SPO = 0, SPH = 0

Временные диаграммы последовательного синхронного обмена в режиме SPI с SPO = 0, SPH = 0 показаны на рисунках: 3.11.4 — одиночный обмен и 3.11.5 — непрерывный обмен.

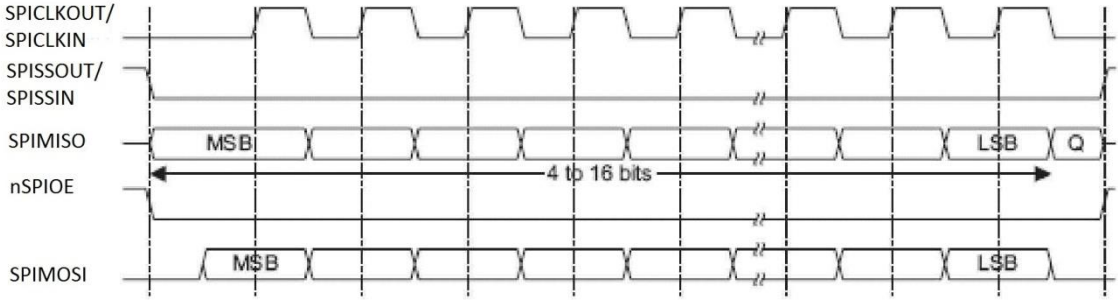


Рисунок 3.11.4 — Формат синхронного обмена протокола SPI, SPO=0, SPH=0 (одиночный обмен)

Примечание — На рисунке буквой Q обозначен сигнал с неопределённым уровнем.

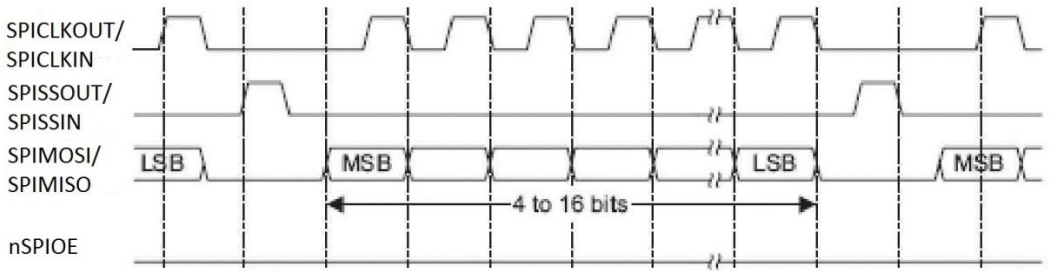


Рисунок 3.11.5 — Формат синхронного обмена протокола SPI, SPO = 0, SPH = 0 (непрерывный обмен)

В данном режиме во время ожидания приёмопередатчика:

- сигнал SPI_SCK имеет низкий логический уровень,
- сигнал SPI_SS имеет высокий логический уровень,
- сигнал SPI_MOSI переводится в высокоимпедансное состояние.

Если функционирование модуля разрешено и в буфере FIFO передатчика содержатся корректные данные, сигнал SPI_SS переводится в низкий логический уровень, что указывает на начало обмена данными и разрешает передачу данных от ведомого устройства на входную линию SPI_MISO ведущего. Контакт передатчика SPIMOSI переходит из высокоимпедансного в активное состояние.

Изм.	Лист	№ докум.	Подп.	Дата
Изм.	Лист	№ докум.	Подп.	Дата
Изм.	Лист	№ докум.	Подп.	Дата
Изм.	Лист	№ докум.	Подп.	Дата
Изм.	Лист	№ докум.	Подп.	Дата

По истечении полутакта сигнала SPI_SCK на линии SPI_MOSI формируется значение первого бита передаваемых данных. К этому моменту должны быть сформированы данные на линиях обмена как ведущего, так и ведомого устройства. По истечении следующего полутакта сигнал SPI_SCK переводится в высокий логический уровень.

Далее данные регистрируются по переднему фронту и выдаются в линию по заднему фронту сигнала SPI_SCK.

В случае передачи одного слова данных после приёма его последнего бита линия SPI_SS переводится в высокий логический уровень по истечении одного периода тактового сигнала SPI_SCK.

В режиме непрерывной передачи данных на линии SPI_SS должны формироваться импульсы высокого логического уровня между передачами каждого из слов данных. Это связано с тем, что в режиме SPH = 0 линия выбора ведомого устройства в низком уровне блокирует запись в сдвиговый регистр. Поэтому ведущее устройство должно переводить линию SPI_SS в высокий уровень по окончании передачи каждого кадра, разрешая таким образом запись новых данных. По окончании приёма последнего бита блока данных линия SPI_SS переводится в состояние, соответствующее режиму ожидания, по истечении одного такта сигнала SPI_SCK.

3.11.6.13 Формат синхронного обмена SPI фирмы Motorola, SPO = 0, SPH = 1

Временные диаграммы последовательного синхронного обмена в режиме SPI с SPO = 0, SPH = 1 содержатся на рисунке 3.11.6 — одиночный и непрерывный обмен.

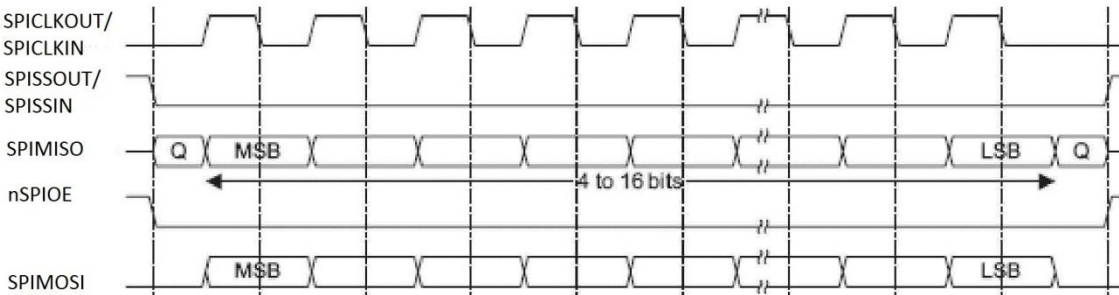


Рисунок 3.11.6 — Формат синхронного обмена протокола SPI, SPO=0, SPH=1

Примечание — На рисунке буквой Q обозначен сигнал с неопределённым уровнем.

В данном режиме во время ожидания приёмопередатчика:

- сигнал SPI_SCK имеет низкий логический уровень;
- сигнал SPI_SS имеет высокий логический уровень;
- сигнал SPI_MOSI переводится в высокоимпедансное состояние.

Если функционирование модуля разрешено и в буфере FIFO передатчика содержатся корректные данные, сигнал SPI_SS переводится в низкий логический уровень, что указывает на начало обмена данными и разрешает передачу данных от ведомого устройства на входную линию SPI_MISO ведущего. Выходной контакт передатчика SPIMOSI переходит из высокоимпедансного в активное состояние.

По истечении полутакта сигнала SPI_SCK на линиях обмена как ведущего, так и ведомого устройств будут сформированы значения первых бит передаваемых данных. В это же время включается линия SPI_SCK и на ней формируется передний фронт сигнала.

Далее данные регистрируются по заднему фронту и выдаются в линию по переднему фронту сигнала SPI_SCK.

В случае передачи одного слова данных после приёма его последнего бита линия SPI_SS переводится в высокий логический уровень по истечении одного периода тактового сигнала SPI_SCK.

В режиме непрерывной передачи данных линия SPI_SS постоянно находится в низком логическом уровне, и переводится в высокий уровень по окончании приёма последнего бита блока данных, как и в режиме передачи одного слова.

3.11.6.14 Формат синхронного обмена SPI фирмы Motorola, SPO = 1, SPH = 0

Временные диаграммы последовательного синхронного обмена в режиме SPI с SPO = 1, SPH = 0 показаны на рисунках: 3.11.7 — одиночный обмен и 3.11.8 — непрерывный обмен.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	<div>Микросхема интегральная 5023BC016</div> <div>Техническое описание</div> <div>Версия 2.04 от 01.09.2017 г.</div>					Лист
Изм	Лист	№ докум.	Подп.	Дата						141

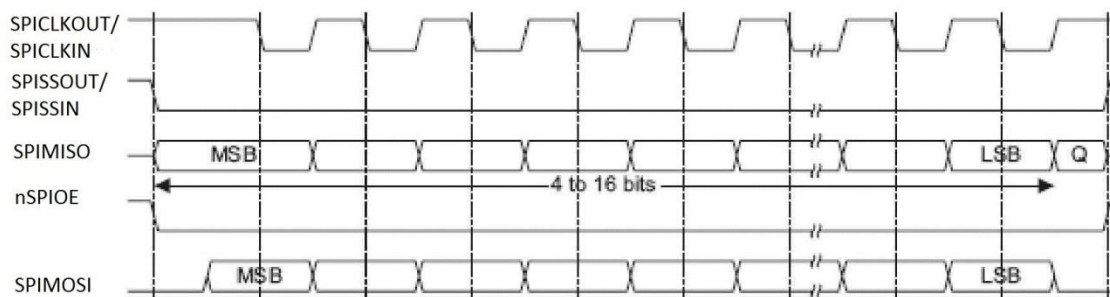


Рисунок 3.11.7 — Формат синхронного обмена протокола SPI, SPO = 1, SPH = 0 (одиначный обмен)

Примечание — На рисунке буквой Q обозначен сигнал с неопределённым уровнем.

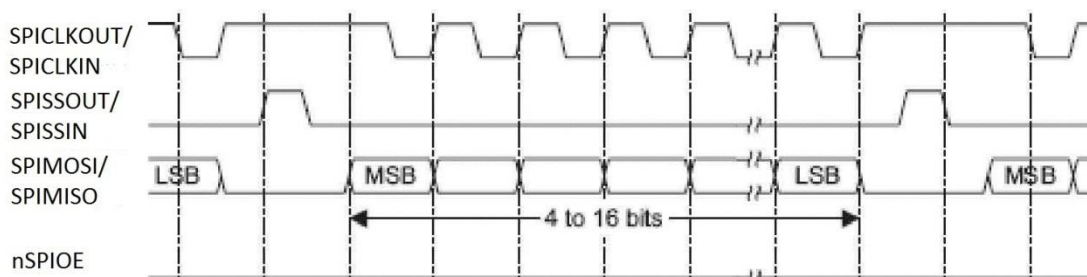


Рисунок 3.11.8 — Формат синхронного обмена протокола SPI, SPO = 1, SPH = 0 (непрерывный обмен)

В данном режиме во время ожидания приёмопередатчика:

- сигнал SPI_SCK имеет высокий логический уровень;
- сигнал SPI_SS имеет высокий логический уровень;
- сигнал SPI_MOSI переводится в высокоимпедансное состояние.

Если функционирование модуля разрешено и в буфере FIFO передатчика содержатся корректные данные, сигнал SPI_SS переводится в низкий логический уровень, что указывает на начало обмена данными и разрешает передачу данных от ведомого устройства на входную линию SPI_MISO ведущего. Выходной контакт передатчика SPIMOSI переходит из высокоимпедансного в активное состояние.

По истечении полутакта сигнала SPI_SCK, на линии SPI_MOSI формируется значение первого бита передаваемых данных. К этому моменту должны быть сформированы данные на линиях обмена как ведущего, так и ведомого устройства. По истечении следующего полутакта сигнал SPI_SCK переводится в низкий логический уровень.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>					Лист
Изм.	Лист	№ докум.	Подп.	Дата						142

Далее данные регистрируются по заднему фронту и выдаются в линию по переднему фронту сигнала SPI_SCK.

В случае передачи одного слова данных после приёма его последнего бита линия SPI_SS переводится в высокий логический уровень по истечении одного периода тактового сигнала SPI_SCK.

В режиме непрерывной передачи данных на линии SPI_SS должны формироваться импульсы высокого логического уровня между передачами каждого из слов данных. Это связано с тем, что в режиме SPH = 0 линия выбора ведомого устройства в низком уровне блокирует запись в сдвиговый регистр. Поэтому ведущее устройство должно переводить линию SPI_SS в высокий уровень по окончании передачи каждого кадра, разрешая таким образом запись новых данных. По окончании приёма последнего бита блока данных линия SPI_SS переводится в состояние, соответствующее режиму ожидания, по истечении одного такта сигнала SPI_SCK.

3.11.6.15 Формат синхронного обмена SPI фирмы Motorola, SPO = 1, SPH = 1

Временные диаграммы последовательного синхронного обмена в режиме SPI с SPO = 1, SPH = 1 показаны на рисунке 3.11.9 — одиночный и непрерывный обмен.

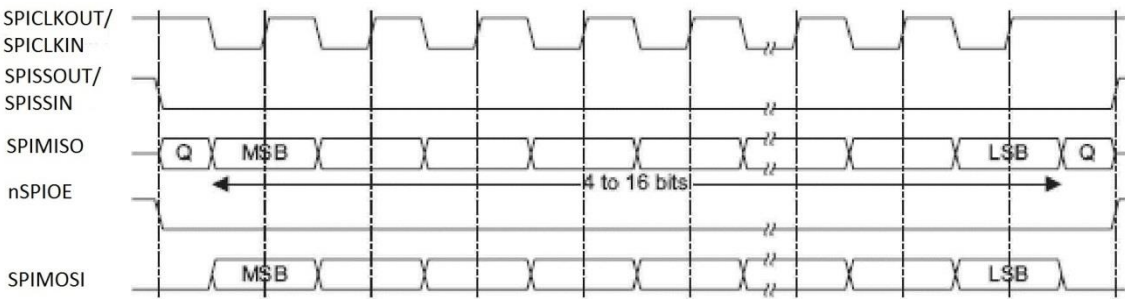


Рисунок 3.11.9 — Формат синхронного обмена протокола SPI, SPO=1, SPH=1

Примечание — На рисунке буквой Q обозначен сигнал с неопределённым уровнем.

В данном режиме во время ожидания приёмопередатчика:

- сигнал SPI_SCK имеет высокий логический уровень;
- сигнал SPI_SS имеет высокий логический уровень;
- сигнал SPI_MOSI переводится в высокоимпедансное состояние.

Если функционирование модуля разрешено и в буфере FIFO передатчика содержатся корректные данные, сигнал SPI_SS переводится в низкий логический уровень, что указывает на начало обмена данными и разрешает передачу данных от ведомого устройства на входную линию SPI_MISO ведущего. Выходной контакт передатчика SPI_MOSI переходит из высокоимпедансного в активное состояние.

По истечении полутакта сигнала SPI_SCK на линиях обмена как ведущего, так и ведомого устройств сформированы значения первых бит передаваемых данных. В это же время включается линия SPI_SCK и на ней формируется передний фронт сигнала.

Далее данные регистрируются по переднему фронту и выдаются в линию по заднему фронту сигнала SPI_SCK.

В случае передачи одного слова данных после приёма его последнего бита линия SPI_SS переводится в высокий логический уровень по истечении одного периода тактового сигнала SPI_SCK.

В режиме непрерывной передачи данных линия SPI_SS постоянно находится в низком логическом уровне и переводится в высокий уровень по окончании приёма последнего бита блока данных, как и в режиме передачи одного слова.

3.11.6.16 Формат синхронного обмена Microwire фирмы National Semiconductor

Временные диаграммы последовательного синхронного обмена в режиме SPI показаны на рисунке 3.11.10 — одиночный обмен и на рисунке 3.11.11 — непрерывный обмен.

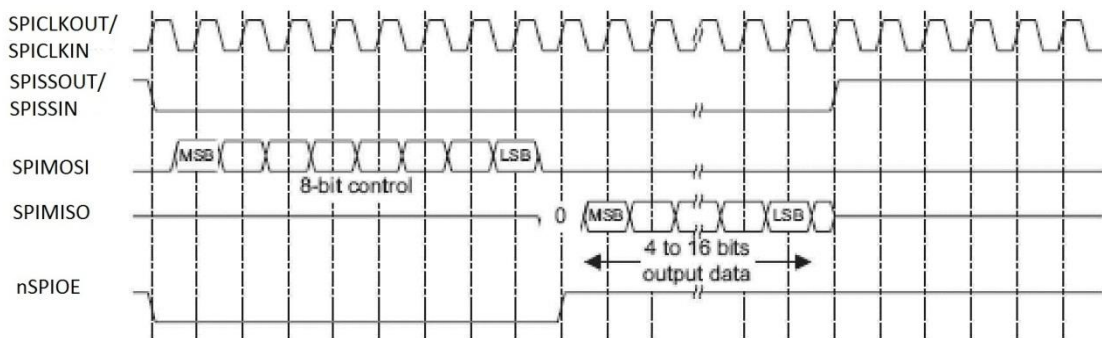


Рисунок 3.11.10 — Формат синхронного обмена протокола Microwire (одиночный обмен)

Протокол передачи данных Microwire во многом схож с протоколом SPI, за исключением того, что обмен в нём осуществляется в полудуплексном режиме, с использованием служебных последовательностей. Каждый информационный обмен начинается с передачи ведущим устройством специальной восьмибитной управляющей последовательности. В течение всего времени ее передачи приёмник не обрабатывает каких-либо входных данных. После того, как сигнал передан и декодирован ведомым устройством, оно выдерживает паузу в один тактовый интервал после передачи последнего бита управляющей последовательности, после чего передаёт в адрес ведущего устройства запрошенные данные. Длительность блока данных от ведомого устройства может составлять от 4 до 16 бит, таким образом, общая длительность информационного кадра составляет от 13 до 25 бит.

В данном режиме во время ожидания приёмопередатчика:

- сигнал SPI_SCK имеет низкий логический уровень;
- сигнал SPI_SS имеет высокий логический уровень;
- сигнал SPI_MOSI переводится в высокоимпедансное состояние.

Переход в режим информационного обмена происходит после записи управляющего байта в буфер FIFO передатчика. По заднему фронту сигнала SPI_SS данные из буфера переносятся в регистр сдвига блока передатчика, откуда, начиная со старшего значащего разряда, последовательно выдаются в линию SPI_MOSI. Линия SPI_SS остается в низком логическом уровне в течение всей передачи кадра. Линия SPI_MISO при этом находится в высокоимпедансном состоянии.

Внешнее ведомое устройство осуществляет приём бит данных по переднему фронту сигнала SPI_SCK. По окончании приёма последнего бита управляющей последовательности она декодируется в течение одного тактового интервала, после чего ведомое устройство передает запрошенные данные в адрес модуля SPI. Биты данных выдаются в линию SPI_MISO по заднему фронту сигнала SPI_SCK. Ведущее устройство, в свою очередь, регистрирует их по переднему фронту этого тактового сигнала. В случае одиночного информационного обмена по окончании приёма последнего бита слова данных сигнал SPI_SS переводится в высокий уровень на

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.						Лист
Изм	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.					145

время, соответствующее одному тактовому интервалу, что служит командой для переноса принятого слова данных из регистра сдвига в буфер FIFO приёмника.

Примечание — Внешнее устройство может перевести линию приёмника в третье состояние по заднему фронту сигнала SPI_SCK после приёма последнего бита слова данных, либо после перевода линии SPI_SS в высокий логический уровень.

Непрерывный обмен данными начинается и заканчивается так же, как и одиночный обмен. Однако линия SPI_SS удерживается в низком логическом уровне в течение всего сеанса передачи данных. Управляющий байт следующего информационного кадра передается сразу же после приёма младшего значащего разряда текущего кадра. Данные из сдвигового регистра передаются в буфер приёмника после регистрации младшего разряда очередного слова по заднему фронту сигнала SPI_SCK.

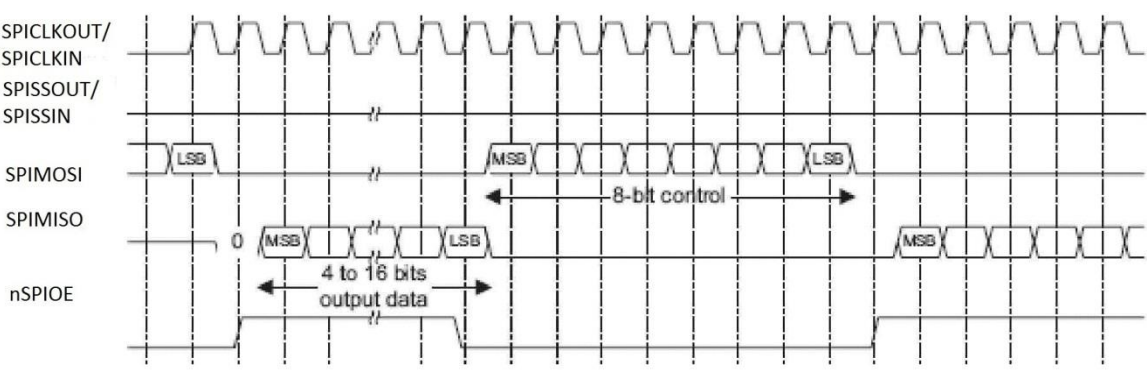


Рисунок 3.11.11 — Формат синхронного обмена протокола Microwire (непрерывный обмен)

3.11.6.17 Требования к временным параметрам сигнала SPI_SS относительно тактового сигнала SPI_SCK в режиме Microwire

Модуль SPI, работающий в режиме Microwire как ведомое устройство, регистрирует данные по переднему фронту сигнала SPI_SCK после установки сигнала SPI_SS в низкий логический уровень. Ведущие устройства, формирующие сигнал SPI_SCK, должны гарантировать достаточное время установки и удержания сигнала SPI_SS по отношению к переднему фронту сигнала SPI_SCK.

Данные требования иллюстрирует рисунок 3.11.12. По отношению к переднему фронту сигнала SPI_SCK, по которому осуществляется регистрация

Изм.	Лист	№ докум.	Подп.	Дата
Изм.	Лист	№ докум.	Подп.	Дата
Изм.	Лист	№ докум.	Подп.	Дата
Изм.	Лист	№ докум.	Подп.	Дата
Изм.	Лист	№ докум.	Подп.	Дата

данных в приёмнике ведомого модуля SPI, время установки сигнала SPI_SS должно быть, как минимум в два раза больше периода SPI_SCK, на котором функционирует модуль. По отношению к предыдущему переднему фронту сигнала SPI_SCK должно обеспечиваться время удержания не менее одного периода этого тактового сигнала.

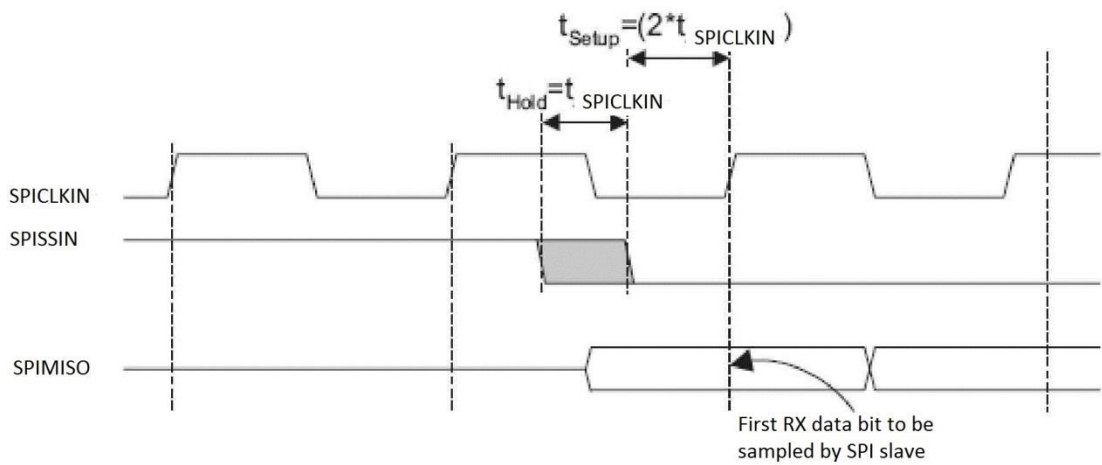


Рисунок 3.11.12 — Формат Microwire, требования к времени установки и удержания сигнала

3.11.7 Интерфейс прямого доступа к памяти

Модуль SPI предоставляет интерфейс подключения к контроллеру прямого доступа к памяти. Функционирование в данном режиме контролируется регистром управления DMASPIDMACR.

Интерфейс DMA включает в себя следующие сигналы для приёма:

- SPIMISOMASREQ — запрос передачи отдельного символа, инициируется приёмопередатчиком. Сигнал переводится в активное состояние в случае, если буфер FIFO приёмника содержит по меньшей мере один символ;
- SPIMISOMABREQ — запрос блочного обмена данными, инициируется модулем приёмопередатчика. Сигнал переходит в активное состояние в случае, если буфер FIFO приёмника содержит четыре или более символов;
- SPIMISOMACLR — сброс запроса на DMA, инициируется контроллером DMA с целью сброса принятого запроса. В случае, если был запрошен блочный обмен данными, сигнал сброса формируется в ходе передачи последнего символа данных в блоке;

Интерфейс DMA включает в себя следующие сигналы для передачи:

— SPIMOSIMASREQ — запрос передачи отдельного символа, инициируется модулем приёмопередатчика. Сигнал переводится в активное состояние в случае, если буфер FIFO передатчика содержит по меньшей мере одну свободную ячейку;

— SPIMOSIMABREQ — запрос блочного обмена данными, инициируется модулем приёмопередатчика. Сигнал переводится в активное состояние в случае, если буфер FIFO передатчика содержит четыре или менее символов;

— SPIMOSIMACLR — сброс запроса на DMA, инициируется контроллером DMA с целью сброса принятого запроса. В случае если был запрошен блочный обмен данными, сигнал сброса формируется в ходе передачи последнего символа данных в блоке.

Сигналы блочного и одноэлементного обмена данными не являются взаимоисключающими, они могут быть инициированы одновременно. Например, в случае, если заполнение данными буфера приёмника превышает пороговое значение четыре, формируются как сигнал запроса одноэлементного обмена, так и сигнал запроса блочного обмена данными. В случае, если количество данных в буфере приёма меньше порогового значения, формируется только запрос одноэлементного обмена. Это бывает полезно в ситуациях, при которых объём данных меньше размера блока. Пусть, например, нужно принять 19 символов. Тогда контроллер DMA осуществит четыре передачи блоков по четыре символа, а оставшиеся три символа передаст в ходе трёх одноэлементных обменов.

Примечание — Для оставшихся трёх символов контроллер SPI не инициирует процедуру блочного обмена.

Каждый инициированный приёмопередатчиком сигнал запроса DMA остаётся активным до момента его сброса соответствующим сигналом DMACLR.

После снятия сигнала сброса модуль приёмопередатчика вновь получает возможность сформировать запрос на DMA в случае выполнения описанных выше условий. Все запросы DMA снимаются после запрета работы приёмопередатчика, а также в случае снятия сигнала разрешения DMA.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	<div>Микросхема интегральная 5023BC016</div> <div>Техническое описание</div> <div>Версия 2.04 от 01.09.2017 г.</div>					Лист
Изм	Лист	№ докум.	Подп.	Дата						148

В таблице 3.11.1 приведены значения порогов заполнения буферов приёмника и передатчика, необходимых для срабатывания запросов блочного обмена DMABREQ.

Таблица 3.11.1 — Параметры срабатывания запросов блочного обмена данными в режиме DMA

Пороговый уровень	Длина блока обмена данными	
	Буфер передатчика (количество незаполненных ячеек)	Буфер приёмника (количество заполненных ячеек)
1/2	4	4

Рисунок 3.11.13 отображает временные диаграммы одноэлементного и блочного запросов DMA, в том числе действие сигнала DMACLR.

Все сигналы должны быть синхронизированы с PCLK.

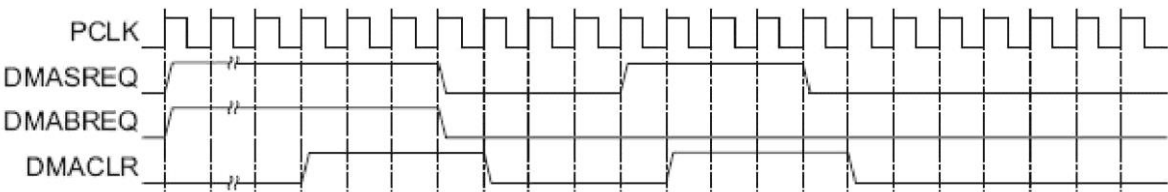


Рисунок 3.11.13 — Временные диаграммы обмена в режиме DMA

3.11.8 Программное управление модулем

3.11.8.1 Общая информация и описание регистров контроллера SPI

В микросхеме реализовано четыре модуля SPI, базовые адреса каждого модуля указаны в таблице 3.11.2.

Смещение адреса каждого регистра относительно базового адреса является постоянным.

Следующие адреса являются резервными и не должны использоваться в нормальном режиме функционирования:

- адреса со смещениями в диапазоне 0x028 – 0x07C и 0xFD0 – 0xFDC зарезервированы для перспективных расширений возможностей модуля;
- адреса со смещениями в диапазоне 0x080 – 0x088 зарезервированы для тестирования.

Данные о регистрах модуля SPI приведены в таблице 3.11.2.

Таблица 3.11.2 — Обобщенные данные о регистрах модуля SPI

Смещение	Наименование	Тип	Значение после сброса	Размер, бит	Описание
0x000	CR0	RW	0x0000	16	Регистр SPIx -> CR0 управления 0
0x004	CR1	RW	0x0	4	Регистр SPIx -> CR1 управления 1
0x008	DR	RW	0x----	16	Буфера FIFO приёмника (чтение) Буфер FIFO передатчика (запись) SPIx -> DR
0x00C	SR	RO	0x03	3	Регистр SPIx -> SR состояния
0x010	CPSR	RW	0x00	8	Регистр SPIx -> CPSR делителя тактовой частоты
0x014	IMSC	RW	0x0	4	Регистр SPIx -> IMSC маски прерывания
0x018	RIS	RO	0x8	4	Регистр SPIx -> RIS состояния прерываний без учёта маскирования
0x01C	MIS	RO	0x0	4	Регистр SPIx -> MIS состояния прерываний с учётом маскирования
0x020	ICR	WO	0x0	4	Регистр SPIx -> ICR сброса прерывания
0x024	DMACR	RW	0x0	2	Регистр SPIx -> DMACR управления прямым доступом к памяти
Примечание — В поле «тип» указан вид доступа к регистру: RW – чтение и запись, RO только чтение, WO – только запись.					

3.11.8.2 Регистр SPIx -> CR0

Регистр SPIx -> CR0 является регистром управления и содержит пять групп битовых полей, предназначенных для управления блоками модуля SPI. Описание регистра представлено в таблице 3.11.3

Таблица 3.11.3 — Описание регистра SPIx -> CR0

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
1	2	3
31 – 16	–	Зарезервировано
15 – 8	SCR	Скорость последовательного обмена. Значение поля SCR используется при формировании тактового сигнала обмена данными. Информационная скорость удовлетворяет соотношению: $F_SPI_SCK / (CPSDVR \cdot (1 + SCR))$, где CPSDVR — четное число в диапазоне от 2 до 254 (см. регистр SPICPSR), а SCR — число от 0 до 255.
7	SPH	Фаза сигнала SPICLKOUT (используется только в режиме обмена SPI фирмы Motorola). См. раздел «Формат SPI фирмы Motorola».

Ине. № подл.	Подп. и дата	Взам. инв. №	Ине. № дубл.	Подп. и дата.					
Изм	Лист	№ докум.	Подп.	Дата					
					Микросхема интегральная 5023BC016				
					Техническое описание				
					Версия 2.04 от 01.09.2017 г.				
					Лист				
					150				

Продолжение таблицы 3.11.3

1	2	3
6	SPO	Полярность сигнала SPISCKOUT (используется только в режиме обмена SPI фирмы Motorola). см. раздел «Формат синхронного обмена SPI фирмы Motorola».
5 – 4	FRF	Формат информационного кадра. 00 — протокол SPI фирмы Motorola; 01 — протокол SSI фирмы Texas Instruments; 10 — протокол Microwire фирмы National Semiconductor; 11 — резерв.
3 – 0	DSS	Размер слова данных: 0000 — резерв 1000 — 9 бит 0001 — резерв 1001 — 10 бит 0010 — резерв 1010 — 11 бит 0011 — 4 бита 1011 — 12 бит 0100 — 5 бит 1100 — 13 бит 0101 — 6 бит 1101 — 14 бит 0110 — 7 бит 1110 — 15 бит 0111 — 8 бит 1111 — 16 бит

3.11.8.3 Регистр SPIx -> CR1

Регистр SPIx -> CR1 является регистром управления и содержит четыре группы битовых полей, предназначенных для управления блоками модуля SPI. Описание регистра представлено в таблице 3.11.4.

Таблица 3.11.4 — Описание регистра SPIx -> CR1

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
1	2	3
15 – 4		Резерв, при чтении результат не определён. При записи следует устанавливать в «0»
3	SOD	Запрет выходных линий в режиме ведомого устройства. Бит используется только в режиме ведомого устройства (MS = 1). Это позволяет организовать двусторонний обмен данными в системах, содержащих одно ведущее и несколько ведомых устройств. Бит SOD следует установить в случае, если данный ведомый модуль SPI не должен в настоящее время осуществлять передачу данных в линию SPI_MOSI. При этом линии обмена данных ведомых устройств можно соединить параллельно. 0 — управление линией SPI_MOSI в ведомом режиме разрешено. 1 — управление линией SPI_MOSI в ведомом режиме запрещено.
2	MS	Выбор ведущего или ведомого режима работы: 0 — режим работы ведущего; 1 — режим работы ведомого.
1	SSE	Разрешение работы приёмопередатчика: 0 — функционирование запрещено; 1 — функционирование разрешено.

Таблица 3.11.4 — Описание регистра SPIx -> CR1					
№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений			
1	2	3			
15 – 4		Резерв, при чтении результат не определён. При записи следует устанавливать в «0»			
3	SOD	Запрет выходных линий в режиме ведомого устройства. Бит используется только в режиме ведомого устройства (MS = 1). Это позволяет организовать двусторонний обмен данными в системах, содержащих одно ведущее и несколько ведомых устройств. Бит SOD следует установить в случае, если данный ведомый модуль SPI не должен в настоящее время осуществлять передачу данных в линию SPI_MOSI. При этом линии обмена данных ведомых устройств можно соединить параллельно. 0 — управление линией SPI_MOSI в ведомом режиме разрешено. 1 — управление линией SPI_MOSI в ведомом режиме запрещено.			
2	MS	Выбор ведущего или ведомого режима работы: 0 — режим работы ведущего; 1 — режим работы ведомого.			
1	SSE	Разрешение работы приёмопередатчика: 0 — функционирование запрещено; 1 — функционирование разрешено.			
Изм.		Лист		№ докум.	
Подп.		Дата			
Микросхема интегральная 5023BC016					Лист
Техническое описание					151
Версия 2.04 от 01.09.2017 г.					

Продолжение таблицы 3.11.4

1	2	3
0	LBM	Тестирование по шлейфу: 0 — нормальный режим функционирования приёмопередатчика; 1 — выход регистра сдвига передатчика соединен со входом регистра сдвига приёмника.

3.11.8.4 Регистр SPIx -> DR

Регистр SPIx -> DR является регистром данных. Регистр SPIDR имеет разрядность 16 бит и предназначен для чтения принятых и записи передаваемых данных.

Операция чтения обеспечивает доступ к последней несчитанной ячейке буфера FIFO приёмника. Запись данных в этот буфер FIFO осуществляет блок приёмника.

Операция записи позволяет занести очередное слово в буфер FIFO передатчика. Извлечение данных из этого буфера осуществляет блок передатчика. При этом извлечённые данные помещаются в регистр сдвига передатчика, откуда последовательно выдаются на линию SPI_MOSI с заданной скоростью информационного обмена.

В случае если выбран размер информационного слова менее 16 бит, перед записью в регистр SPIDR необходимо обеспечить выравнивание данных по правой границе. Блок передатчика игнорирует неиспользуемые биты. Принятые информационные слова автоматически выравниваются по правой границе в блоке приёмника.

В режиме обмена данными Microwire фирмы National Semiconductor модуль SPI по умолчанию работает с восьмиразрядными информационными словами (старший значащий байт игнорируется). Размер принимаемых данных задаётся программно. Буфера FIFO приёмника и передатчика автоматически не очищаются даже в случае, если бит SSE установлен в «0». Это позволяет заполнить буфер передатчика необходимой информацией заблаговременно, перед разрешением работы модуля. Описание регистра представлено в таблице 3.11.5.

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	информационного обмена.					
					В случае если выбран размер информационного слова менее 16 бит, перед записью в регистр SPIDR необходимо обеспечить выравнивание данных по правой границе. Блок передатчика игнорирует неиспользуемые биты. Принятые информационные слова автоматически выравниваются по правой границе в блоке приёмника.					
					В режиме обмена данными Microwire фирмы National Semiconductor модуль SPI по умолчанию работает с восьмиразрядными информационными словами (старший значащий байт игнорируется). Размер принимаемых данных задаётся программно. Буфера FIFO приёмника и передатчика автоматически не очищаются даже в случае, если бит SSE установлен в «0». Это позволяет заполнить буфер передатчика необходимой информацией заблаговременно, перед разрешением работы модуля. Описание регистра представлено в таблице 3.11.5.					
					Микросхема интегральная 5023BC016					Лист
					Техническое описание					
					Версия 2.04 от 01.09.2017 г.					
Изм.	Лист	№ докум.	Подп.	Дата						152

Таблица 3.11.5 — Описание регистра DR

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
15 – 0	DATA	Принимаемые данные (чтение). Передаваемые данные (запись). В случае если выбран размер информационного слова менее 16 бит, перед записью в регистр SPIDR необходимо обеспечить выравнивание данных по правой границе. Блок передатчика игнорирует неиспользуемые биты. Принятые информационные слова автоматически выравниваются по правой границе в блоке приёмника.

3.11.8.5 Регистр SPIx -> SR

Регистр SPIx -> SR является регистром состояния и доступен только для чтения и содержит информацию о состоянии буферов FIFO приёмника и передатчика, и занятости модуля SPI. Описание регистра SPIx -> SR представлено в таблице 3.11.6.

Таблица 3.11.6 — Описание регистра SPIx -> SR

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
15 – 5	—	
4	BSY	Флаг активности модуля: 0 — модуль SPI не активен; 1 — модуль SPI в настоящее время передаёт и/или принимает данные, либо буфер FIFO передатчика не пуст.
3	RFF	Буфер FIFO приёмника заполнен: 0 — не заполнен; 1 — заполнен.
2	RNE	Буфер FIFO приёмника не пуст: 0 — пуст; 1 — не пуст.
1	TNF	Буфер FIFO передатчика не заполнен: 0 — заполнен; 1 — не заполнен.
0	TFE	Буфер FIFO передатчика пуст: 0 — не пуст; 1 — пуст.

3.11.8.6 Регистр SPIx -> CPSR

Регистр SPIx -> CPSR является регистром делителя тактовой частоты. Регистр SPIx -> CPSR используется для установки параметров делителя тактовой частоты. Записываемое значение должно быть целым числом в диапазоне от 2 до 254.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>					Лист
Изм.	Лист	№ докум.	Подп.	Дата						153

Младший значащий разряд регистра принудительно устанавливается в ноль. Если записать в регистр SPICPSR нечетное число, его последующее чтение даст результатом это число, но с установленным в ноль младшим битом. Описание регистра SPIx -> CPSR представлено в таблице 3.11.7.

Таблица 3.11.7 — Описание регистра SPIx -> CPSR

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 8	–	
1 – 0	CPSDVSR	Коэффициент деления тактовой частоты. Записываемое значение должно быть целым числом в диапазоне от 2 до 254. Младший значащий разряд регистра принудительно устанавливается в ноль

3.11.8.7 Регистр SPIx -> IMSC

Регистр SPIx -> IMSC предназначен для установки и сброса маски прерывания. При чтении выдаётся текущее значение маски. При записи производится установка или сброс маски на соответствующее прерывание. При этом запись «1» в разряд разрешает соответствующее прерывание, запись «0» — запрещает.

После сброса все биты регистра маски устанавливаются в нулевое состояние. Описание регистра SPIx -> IMSC представлено в таблице 3.11.8.

Таблица 3.11.8 — Описание регистра SPIx -> IMSC

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 4	–	
3	TXIM	Маска прерывания по заполнению на 50% и менее буфера FIFO передатчика. 1 — не маскирована. 0 — маскирована
2	RXIM	Маска прерывания по заполнению на 50% и менее буфера FIFO приёмника. 1 — не маскирована. 0 — маскирована
1	RTIM	Маска прерывания по таймауту приёмника (буфер FIFO приёмника не пуст и не было попыток его чтения в течение времени таймаута). 1 — не маскирована. 0 — маскирована
0	RORIM	Маска прерывания по переполнению буфера приёмника. 1 — не маскирована. 0 — маскирована

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.
Изм	Лист	№ докум.	Подп.	Дата
				Микросхема интегральная 5023BC016
				Техническое описание
				Версия 2.04 от 01.09.2017 г.
				Лист
				154

3.11.8.8 Регистр SPIx -> RIS

Регистр SPIx RIS — регистр состояния прерываний. Этот регистр доступен только для чтения и содержит текущее состояние прерываний без учёта маскирования. Данные, записываемые в регистр, игнорируются. Описание регистра SPIx -> RIS представлено в таблице 3.11.9.

Таблица 3.11.9 — Описание регистра SPIx -> RIS

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 4	–	
3	TXRIS	Состояние до маскирования прерывания SPITXINTR
2	RXRIS	Состояние до маскирования прерывания SPIRXINTR
1	RTRIS	Состояние до маскирования прерывания SPIRTINTR
0	RORRIS	Состояние до маскирования прерывания SPIRORINTR

3.11.8.9 Регистр SPIx -> MIS

Регистр SPIx -> MIS - регистр маскированного состояния прерываний. Этот регистр доступен только для чтения и содержит текущее состояние прерываний с учетом маскирования. Данные, записываемые в регистр, игнорируются. Описание регистра SPIx -> MIS представлено в таблице 3.11.10.

Таблица 3.11.10 — Описание регистра SPIx -> MIS

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 4	–	
3	TXMIS	Состояние маскированного прерывания SPITXINTR
2	RXMIS	Состояние маскированного прерывания SPIRXINTR
1	RTMIS	Состояние маскированного прерывания SPIRTINTR
0	RORMIS	Состояние маскированного прерывания SPIRORINTR

3.11.8.10 Регистр SPIx -> ICR

Регистр SPIx -> ICR предназначен для сброса прерываний. Этот регистр доступен только для записи и предназначен для сброса признака прерывания по заданному событию путём записи «1» в соответствующий бит. Запись в любой из

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	Микросхема интегральная 5023BC016					Лист
					Техническое описание					
					Версия 2.04 от 01.09.2017 г.					155
					Изм	Лист	№ докум.	Подп.	Дата	

разрядов регистра «0» игнорируется. Описание регистра SPIx -> ICR представлено в таблице 3.11.11.

Таблица 3.11.11 — Описание регистра SPIx -> ICR

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 2	–	
1	RTIC	Сброс прерывания SPIRTINTR
0	RORIC	Сброс прерывания SPIRORINTR

3.11.8.11 Регистр SPIx -> DMACR

Регистр SPIx -> DMACR предназначен для управления прямым доступом к памяти. Регистр доступен по чтению и записи. После сброса все биты регистра SPIx -> DMACR обнуляются. Описание регистра SPIx -> DMACR представлено в таблице 3.11.12.

Таблица 3.11.12 — Описание регистра SPIx -> DMACR

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 2		Резерв. Не модифицируется. При чтении выдаются нули.
1	TXDMAE	Использование DMA при передаче. Если бит установлен в 1, разрешено формирование запросов DMA для обслуживания буфера FIFO передатчика
0	RXDMAE	Использование DMA при приёме. Если бит установлен в 1, разрешено формирование запросов DMA для обслуживания буфера FIFO приёмника

3.11.9 Прерывания модуля

В модуле предусмотрено пять маскируемых линий запроса на прерывание с выводом на один общий сигнал, представляющий собой комбинацию независимых по схеме ИЛИ.

Сигналы запроса на прерывание:

— SPIRXINTR — запрос на обслуживание буфера FIFO приёмника. Прерывание по заполнению буфера FIFO приёмника формируется в случае, если буфер приёмника содержит четыре или более несчитанных слов данных;

— SPITXINTR — запрос на обслуживание буфера FIFO передатчика. Прерывание по заполнению буфера FIFO передатчика формируется в случае, если буфер передатчика содержит четыре или менее корректных слов данных. Состояние

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>					Лист
Изм.	Лист	№ докум.	Подп.	Дата						156

прерывания не зависит от значения сигнала разрешения работы модуля SPI. Это позволяет организовать взаимодействие программного обеспечения с передатчиком одним из двух способов. Во-первых, можно записать данные в буфер заблаговременно, перед активизацией передатчика и разрешения прерываний. Во-вторых, можно предварительно разрешить работу модуля и формирование прерываний и заполнять буфер передатчика в ходе работы процедуры обслуживания прерываний;

— SPIRORINTR — переполнение буфера FIFO приёмника. Прерывание по переполнению буфера FIFO приёмника формируется в случае, если буфер уже заполнен и блоком приёмника осуществлена попытка записать в него еще одно слово. При этом принятое слово данных регистрируется в регистре сдвига приёмника, но в буфер приёмника не заносится;

— SPIRTINTR — таймаут приёмника. Прерывание по таймауту приёмника возникает в случае, если буфер FIFO приёмника не пуст, и на вход приёмника не поступало новых данных в течение периода времени, необходимого для передачи 32 бит. Данный механизм гарантирует, что пользователь будет знать о наличии в буфере приёмника необработанных данных. Прерывание по таймауту снимается либо после считывания данных из буфера приёмника до его опустошения, либо после приёма новых слов данных по входной линии SPI_MISO. Кроме того, оно может быть снято путём записи «1» в бит RTIC регистра сброса прерывания SPITICR;

— SPIINTR — логическое ИЛИ сигналов SPIRXINTR, SPITXINTR, SPIRTINTR и SPIRORINTR. Все описанные сигналы запроса на прерывание скомбинированы в общую линию путем объединения по схеме ИЛИ сигналов SPIRXINTR, SPITXINTR, SPIRTINTR и SPIRORINTR с учетом маскирования. Общий выход может быть подключен к системному контроллеру прерывания, что позволит ввести дополнительное маскирование запросов на уровне периферийных устройств.

Каждый из независимых сигналов запроса на прерывание может быть маскирован путём установки соответствующего бита в регистре маски SPIIMSC.

Име. № подл.	Подп. и дата		Име. № дубл.		Подп. и дата					
Изм	Взам. инв. №		Подп. и дата							
Микросхема интегральная 5023BC016						Лист				
Техническое описание						157				
Версия 2.04 от 01.09.2017 г.										
Изм	Лист	№ докум.	Подп.	Дата						

32 бита. Данный механизм гарантирует, что пользователь будет знать о наличии в буфере приёмника необработанных данных. Прерывание по таймауту снимается либо после считывания данных из буфера приёмника до его опустошения, либо после приёма новых слов данных по входной линии SPI_MISO. Кроме того, оно может быть снято путём записи «1» в бит RTIC регистра сброса прерывания SPITICR;

— SPIINTR — логическое ИЛИ сигналов SPIRXINTR, SPITXINTR, SPIRTINTR и SPIRORINTR. Все описанные сигналы запроса на прерывание скомбинированы в общую линию путем объединения по схеме ИЛИ сигналов SPIRXINTR, SPITXINTR, SPIRTINTR и SPIRORINTR с учетом маскирования. Общий выход может быть подключен к системному контроллеру прерывания, что позволит ввести дополнительное маскирование запросов на уровне периферийных устройств.

Каждый из независимых сигналов запроса на прерывание может быть маскирован путём установки соответствующего бита в регистре маски SPIIMSC.

Установка бита в «1» разрешает соответствующее прерывание, а в «0» — запрещает.

Доступность индивидуальных линий и общей линии запроса позволяет организовать обслуживание прерываний в системе как путём применения глобальной процедуры обработки, так и с помощью драйвера устройства, построенного по модульному принципу.

Прерывания от приёмника и передатчика SPIRXINTR и SPITXINTR выведены отдельно от прерываний по изменению состояния устройства. Это позволяет использовать данные сигналы запроса для обеспечения чтения и записи данных согласованно с достижением заданного порога заполнения буферов FIFO приёмника и передатчика.

Признаки возникновения каждого из условий прерывания можно считать либо из регистра прерываний SPIRIS, либо из маскированного регистра прерываний SPIMIS.

3.12 Контроллер интерфейса UART

3.12.1 Общие положения

Модуль универсального асинхронного приёмопередатчика UART представляет собой периферийное устройство микросхемы.

3.12.2 Основные характеристики интерфейса UART

3.12.2.1 Общие характеристики

Модуль UART содержит независимые буферы приёма (16 x 12) и передачи (16 x 8) типа FIFO, что позволяет снизить интенсивность прерываний центрального процессора.

Программное отключение FIFO позволяет ограничить размер буфера одним байтом.

Программное управление скоростью обмена. Обеспечивается возможность деления тактовой частоты опорного генератора в диапазоне (1 x 16 – 65535 x 16). Допускается использование нецелых коэффициентов деления частоты, что позволяет использовать любой опорный генератор с частотой более 3,6864 МГц.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	3.12.1 Общие положения					
					Модуль универсального асинхронного приёмопередатчика UART представляет собой периферийное устройство микросхемы.					
					3.12.2 Основные характеристики интерфейса UART					
					3.12.2.1 Общие характеристики					
Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	Модуль UART содержит независимые буферы приёма (16 x 12) и передачи (16 x 8) типа FIFO, что позволяет снизить интенсивность прерываний центрального процессора.					
					Программное отключение FIFO позволяет ограничить размер буфера одним байтом.					
					Программное управление скоростью обмена. Обеспечивается возможность деления тактовой частоты опорного генератора в диапазоне (1 x 16 – 65535 x 16).					
					Допускается использование нецелых коэффициентов деления частоты, что позволяет использовать любой опорный генератор с частотой более 3,6864 МГц.					
Изм.	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016					Лист
					Техническое описание					
					Версия 2.04 от 01.09.2017 г.					158

Поддержка стандартных элементов асинхронного протокола связи – стартового и стопового бит, а также бита контроля чётности, которые добавляются перед передачей и удаляются после приёма.

Независимое маскирование прерываний от буфера FIFO передатчика, буфера FIFO приёмника, по таймауту приёмника, по изменению линий состояния модема, а также в случае обнаружения ошибки.

Поддержка прямого доступа к памяти.

Обнаружение ложных стартовых бит.

Формирование и обнаружения сигнала разрыва линии.

Возможность организации аппаратного управления потоком данных.

Полностью программируемый асинхронный последовательный интерфейс с характеристиками:

- данные длиной 5, 6, 7 или 8 бит;
- формирование и контроль четности (проверочный бит выставляется по чётности, нечётности, имеет фиксированное значение, либо не передаётся);
- формирование 1 или 2 стоповых бит;
- скорость передачи данных – от 0 до UARTCLK/16 Бод.

Наличие идентификационного регистра, однозначно идентифицирующего модуль, что позволяет операционной системе выполнять автоматическую конфигурацию.

3.12.2.2 Программируемые параметры

Следующие ключевые параметры могут быть заданы программно:

- скорость передачи данных – целая и дробная часть числа;
- количество бит данных;
- количество стоповых бит;
- режим контроля чётности;
- разрешение или запрет использования буферов FIFO (глубина очереди данных — 32 элемента или один элемент, соответственно);

Име. № подл.	Подп. и дата	Взам. име. №	Име. № дубл.	Подп. и дата.	<div>Микросхема интегральная 5023BC016</div> <div>Техническое описание</div> <div>Версия 2.04 от 01.09.2017 г.</div>					Лист
Изм	Лист	№ докум.	Подп.	Дата						159

— порог срабатывания прерывания по заполнению буферов FIFO (1/8, 1/4, 1/2, 3/4 и 7/8);

— частота внутреннего тактового генератора (номинальное значение — 1,8432 МГц) может быть задана в диапазоне 1,42 – 2,12 МГц для обеспечения возможности формирования бит данных с укороченной длительностью в режиме пониженного энергопотребления;

— режим аппаратного управления потоком данных.

3.12.2.3 Отличия от контроллера UART 16C650

Контроллер отличается от промышленного стандарта асинхронного приёмопередатчика 16C650 следующими характеристиками:

— пороги срабатывания прерывания по заполнению буфера FIFO приёмника — 1/8, 1/4, 1/2, 3/4 и 7/8;

— пороги срабатывания прерывания по заполнению буфера FIFO передатчика — 1/8, 1/4, 1/2, 3/4 и 7/8;

— отличается распределение адресов внутренних регистров и назначение бит в регистрах;

— недоступны изменения сигналов о состоянии модема.

Следующие возможности контроллера 16C650 не поддерживаются:

— полуторная длительность стопового бита (поддерживается только 1 или 2 стоповых бита);

— независимое задание тактовой частоты приёмника и передатчика.

3.12.2.4 Функциональные возможности

Устройство выполняет следующие функции:

— преобразование данных, полученных от периферийного устройства, из последовательной в параллельную форму;

— преобразование данных, передаваемых на периферийное устройство, из параллельной в последовательную форму.

Процессор читает и записывает данные, а также управляющую информацию и информацию о состоянии модуля. Приём и передача данных буферизуются с

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	<div>Микросхема интегральная 5023BC016</div> <div>Техническое описание</div> <div>Версия 2.04 от 01.09.2017 г.</div>					Лист 160	
Изм.	Лист	№ докум.	Подп.	Дата							

помощью внутренней памяти FIFO, позволяющей сохранить до 16 байт независимо для режимов приёма и передачи.

Модуль приёмопередатчика:

— содержит программируемый генератор, формирующий тактовый сигнал одновременно для передачи и для приёма данных на основе внутреннего тактового сигнала UARTCLK;

— обеспечивает возможности, сходные с возможностями промышленного стандарта – контроллера UART 16C650;

— позволяет осуществлять обмен информацией в режиме UART с максимальной скоростью до 921600 бит/с.

Режим работы приёмопередатчика и скорость обмена данными контролируются регистром управления линией UARTLCR_N и регистрами делителя скорости передачи данных — целой части (UARTIBRD) и дробной части (UARTFBRD).

Устройство может формировать следующие сигналы:

— независимые маскируемые прерывания от приёмника (в том числе по таймауту), передатчика;

— общее прерывание, возникающее в случае, если возникло одно из независимых немаскированных прерываний;

— сигналы запроса на прямой доступ к памяти (DMA) для совместной работы с контроллером DMA.

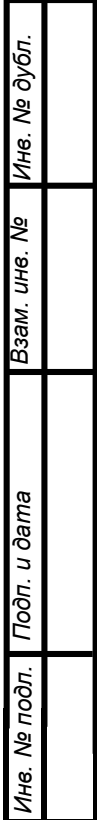
В случае возникновения ошибки в структуре сигнала, чётности данных, а также разрыва линии соответствующий бит ошибки устанавливается и сохраняется в буфере FIFO. В случае переполнения буфера немедленно устанавливается соответствующий бит в регистре переполнения, а доступ к записи в буфер FIFO блокируется.

Существует возможность программно ограничить размер буфера FIFO одним байтом, что позволяет реализовать общепринятый интерфейс асинхронной последовательной связи с двойной буферизацией.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	<div>Микросхема интегральная 5023BC016</div> <div>Техническое описание</div> <div>Версия 2.04 от 01.09.2017 г.</div>					Лист
Изм.	Лист	№ докум.	Подп.	Дата						161

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.



Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.

3.12.3.2 Буфер FIFO передатчика

Буфер передатчика имеет разрядность 8 бит, глубину 32 слова, схему организации доступа типа FIFO.

Данные от центрального процессора, записанные через шину APB, сохраняются в буфере до тех пор, пока не будут считаны логической схемой передачи данных.

Существует возможность запретить буфер FIFO передатчика, в этом случае он будет функционировать как однобайтовый буферный регистр.

3.12.3.3 Буфер FIFO приёмника

Буфер приёмника имеет ширину 12 бит, глубину 32 слова, схему организации доступа типа FIFO.

Принятые от периферийного устройства данные и соответствующие кодов ошибки сохраняются логикой приёма данных в нём до тех пор, пока не будут считаны центральным процессором через шину APB.

Буфер FIFO приёмника может быть запрещён, в этом случае он будет действовать как однобайтовый буферный регистр.

3.12.3.4 Блок передатчика

Логические схемы передатчика осуществляют преобразование данных, считанных из буфера передатчика, из параллельной в последовательную форму.

Управляющая логика выдаёт последовательный поток бит в порядке: стартовый бит, биты данных, начиная с младшего значащего разряда, бит проверки на чётность, и, наконец, стоповые биты, в соответствии с конфигурацией, записанной в регистре управления.

3.12.3.5 Блок приёмника

Логические схемы приёмника преобразуют данные, полученные от периферийного устройства, из последовательной в параллельную форму после обнаружения корректного стартового импульса.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.						
Изм	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016					Лист
					Техническое описание					
					Версия 2.04 от 01.09.2017 г.					163

Производится проверка переполнения буфера, проверки на ошибки контроля чётности, на ошибки в структуре сигнала, а также на разрыв линии. Признаки обнаружения этих ошибок также сохраняются в выходном буфере.

3.12.3.6 Блок формирования прерываний

Блок генерирует независимые маскируемые прерывания с активным высоким уровнем. Кроме того, формируется комбинированное прерывание путём объединения указанных независимых прерываний по схеме «ИЛИ».

Комбинированный сигнал прерывания может быть подан на внешний контроллер прерываний системы, при этом появится дополнительная возможность маскирования устройства в целом, что облегчает построение модульных драйверов устройств.

Другой подход состоит в подаче на системный контроллер прерываний независимых линий запроса на прерывание от приёмопередатчика. В этом случае процедура обработки сможет одновременно считать информацию обо всех источниках прерывания. Данный подход привлекателен в случае, если скорость доступа к регистрам периферийных устройств значительно превышает тактовую частоту центрального процессора в системе реального времени.

3.12.3.7 Модуль обеспечения прямого доступа к памяти

Модуль обеспечивает интерфейс с контроллером DMA согласно схеме взаимодействия приёмопередатчика и контроллера DMA.

3.12.3.8 Блок и регистры синхронизации

Контроллер поддерживает как асинхронный, так и синхронный режимы работы тактовых генераторов CPU_CLK и UARTCLK. Регистры синхронизации и логика квитирования постоянно находятся в активном состоянии. Это практически не отражается на характеристиках устройства и занимаемой площади.

Синхронизация сигналов управления осуществляется в обоих направлениях потока данных, то есть как из области действия CPU_CLK в область действия UARTCLK, так и наоборот.

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>					Лист
Изм.	Лист	№ докум.	Подп.	Дата						164

3.12.4 Описание функционирования модуля UART

3.12.4.1 Сброс модуля

Приёмопередатчик и энкодер/декодер могут быть сброшены общим сигналом сброса процессора. Значения регистров после сброса описаны в разделе «Программное управление модулем».

3.12.4.2 Тактовые сигналы

Частота тактового сигнала UARTCLK должна обеспечивать поддержку требуемого диапазона скоростей передачи данных

$$F_UARTCLK(min) \geq 16 \cdot baud_rate_max, \quad (4)$$

$$F_UARTCLK(max) \leq 16 \cdot 65535 \cdot baud_rate_min. \quad (5)$$

Например, для поддержки скорости передачи данных в диапазоне от 110 до 460800 Бод частота UARTCLK должна находиться в интервале от 7,3728 МГц до 115,34 МГц.

Частота UARTCLK, кроме того, должна выбираться с учётом возможности установки скорости передачи данных в рамках заданных требований точности.

Существует ограничение на соотношение между тактовыми частотами CPU_CLK и UARTCLK. Частота UARTCLK должна быть не более, чем в 5/3 раз выше частоты CPU_CLK

$$F_UARTCLK \leq 5/3 \cdot F_CPU_CLK. \quad (6)$$

Например, при работе в режиме UART с максимальной скоростью передачи данных 921600 бод, при частоте UARTCLK 14,7456 МГц, частота CPU_CLK должна быть не менее 8,85276 МГц. Это гарантирует, что контроллер UART будет иметь достаточно времени для записи принятых данных в буфер FIFO.

В реализации контроллера в процессоре на вход модуля UART (UARTCLK) подаётся системная частота.

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	<div>Микросхема интегральная 5023BC016</div> <div>Техническое описание</div> <div>Версия 2.04 от 01.09.2017 г.</div>					Лист
Изм.	Лист	№ докум.	Подп.	Дата						165

3.12.4.3 Функционирование универсального асинхронного приёмопередатчика

Управляющая информация хранится в регистре управления линией UARTLCR. Этот регистр имеет внутреннюю ширину 30 бит, однако внешний доступ по шине APB к нему осуществляется через следующие регистры:

— UARTLCR_H — определяет параметры передачи данных, длину слова, режим буферизации, количество передаваемых стоповых бит, режим контроля четности, формирование сигнала разрыва линии;

— UARTIBRD — определяет целую часть коэффициента деления для скорости передачи данных;

— UARTFBRD — определяет дробную часть коэффициента деления для скорости передачи данных.

3.12.4.4 Коэффициент деления частоты

Коэффициент деления для формирования скорости передачи данных состоит из 22 бит, при этом 16 бит выделено для представления его целой части, а 6 бит — дробной части. Возможность задания нецелых коэффициентов деления позволяет осуществлять обмен данными со стандартными информационными скоростями, при этом используя в качестве UARTCLK тактовый сигнал с произвольной частотой более 3,6864 МГц.

Целая часть коэффициента деления записывается в 16-битный регистр UARTIBRD. Шестиразрядная дробная часть записывается в регистр UARTFBRD. Значение коэффициента деления связано с содержимым указанных регистров следующим образом:

Коэффициент деления

$$UARTCLK / (16 \cdot \text{скорость передачи данных}) = BRD_I + BRD_F, \quad (7)$$

где BRD_I — целая часть;

BRD_F — дробная часть коэффициента деления.

Формат представления коэффициента деления представлен на рисунке 3.12.2.

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>					Лист
Изм.	Лист	№ докум.	Подп.	Дата						166



Рисунок 3.12.2 — Формат представления коэффициента деления

Шестибитное значение, записываемое в регистр UARTFBRD, вычисляется путём выделения дробной части требуемого коэффициента деления, умножения ее на 64 (то есть на $2n$, где n — ширина регистра UARTFBRD) и округления до ближайшего целого числа

$$M = \text{integer} (BRD_F \cdot 2n + 0,5), \quad (8)$$

где $\text{integer} (.)$ — операция отсечения дробной части числа, $n = 6$.

В модуле формируется внутренний сигнал Baud16, представляющий собой последовательность импульсов с длительностью, равной периоду сигнала UARTCLK и средней частотой, в 16 раз большей требуемой скорости обмена данными.

3.12.4.5 Передача и приём данных

Принятые или передаваемые данные записываются в 16-элементные буферы FIFO, при этом каждый элемент приёмного буфера FIFO кроме байта данных хранит также четыре бита информации о состоянии модема.

Для передачи данные заносятся в буфер FIFO передатчика. Если функционирование приёмопередатчика разрешено, начинается передача информационного кадра с параметрами, указанными в регистре управления линией UARTLCR_H. Передача данных продолжается до опустошения буфера FIFO передатчика. После записи элемента в буфер FIFO передатчика сигнал BUSY переходит в высокое состояние. Это состояние сохраняется в течение всего времени передачи данных. В низкое состояние сигнал BUSY переходит только после того, как буфер FIFO передатчика станет пуст, а последний бит данных (включая стоповые биты) будет передан. Сигнал BUSY может находиться в высоком состоянии даже в случае, если приёмопередатчик будет переведён из разрешённого состояния в запрещённое.

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>					Лист
Изм.	Лист	№ докум.	Подп.	Дата						167

Для каждого бита данных (в приёмной линии) производится три измерения уровня, решение принимается по мажоритарному принципу.

В случае если приёмник находился в неактивном состоянии (на линии входного сигнала UART_RXD постоянно присутствовала единица) и произошёл переход входного сигнала из высокого в низкий логический уровень (обнаружен стартовый бит), включается счётчик, тактируемый сигналом Baud16, после чего отсчёты сигнала на входе приёмника регистрируются каждые восемь тактов (в режиме асинхронного приёмопередатчика) сигнала Baud16.

Стартовый бит считается достоверным в случае, если сигнал на линии UART_RXD сохраняет низкий логический уровень в течение восьми отсчётов сигнала Baud16 с момента включения счётчика. В противном случае переход в ноль рассматривается как ложный старт и игнорируется.

В случае если обнаружен достоверный стартовый бит, производится регистрация последовательности данных на входе приёмника. Очередной бит данных фиксируются каждые 16 отсчётов тактового сигнала Baud16 (что соответствует длительности одного символа). Производится регистрация всех бит данных (согласно запрограммированным параметрам) и бита чётности (если включен режим контроля чётности).

В завершение производится проверка присутствия корректного стопового бита (высокий логический уровень сигнала UART_RXD). В случае если последнее условие не выполняется, устанавливается признак ошибки формирования кадра. После того, как слово данных принято полностью, оно заносится в буфер FIFO приёмника, наряду с четырьмя битами признаков ошибки, связанных с принятым словом (таблица 3.12.1).

3.12.4.6 Биты ошибки

Три бита признаков ошибки, ассоциированные с принятым символом данных, заносятся в разряды [10...8] слова данных в буфере FIFO приёмника. Также предусмотрен признак ошибки переполнения буфера FIFO в разряде 11 слова данных.

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>					Лист
Изм.	Лист	№ докум.	Подп.	Дата						168

Таблица 3.12.1 содержит назначение всех бит слова данных в FIFO-буфере приёмника.

Таблица 3.12.1 — Описание слова данных в FIFO-буфере приёмника

Бит буфера FIFO	Назначение
11	Признак переполнения буфера
10	Ошибка – «разрыв линии»
9	Ошибка проверки на чётность
8	Ошибка формирования кадра
7 – 0	Принятые данные

3.12.4.7 Бит переполнения буфера

Бит переполнения непосредственно не связан с конкретным символом в буфере приёмника. Признак переполнения фиксируется в случае, если буфер FIFO заполнен к моменту, когда очередной символ данных полностью принят (находится в регистре сдвига). При этом данные из регистра сдвига не попадают в буфер приёмника и теряются с началом приёма очередного символа. Как только в буфере приёмника появляется свободное место, очередной принятый символ данных заносится в буфер FIFO вместе с текущим значением признака переполнения. После успешной записи данных в буфер признак переполнения сбрасывается.

3.12.4.8 Запрет буфера FIFO

Предусмотрена возможность отключения FIFO буферов приёмника и передатчика. В этом случае приёмная и передающая сторона контроллера UART располагают лишь однобайтными буферными регистрами. Бит переполнения буфера устанавливается при этом тогда, когда очередной символ данных уже принят, однако предыдущий еще не был считан.

В настоящей реализации модуля буферы FIFO физически не отключаются, необходимая функциональность достигается за счёт логических манипуляций с флагами. При этом в случае, если буфер FIFO отключен, а сдвиговый регистр передатчика пуст (не используется), запись байта данных происходит непосредственно в регистр сдвига, минуя буферный регистр.

Изм.	Лист	№ докум.	Подп.	Дата	<div>Микросхема интегральная 5023BC016</div> <div>Техническое описание</div> <div>Версия 2.04 от 01.09.2017 г.</div>		<div>Лист</div> <div>169</div>	
Изм.	Лист	№ докум.	Подп.	Дата	<div>приёмника появляется свободное место, очередной принятый символ данных заносится в буфер FIFO вместе с текущим значением признака переполнения. После успешной записи данных в буфер признак переполнения сбрасывается.</div> <div>3.12.4.8 Запрет буфера FIFO</div> <div>Предусмотрена возможность отключения FIFO буферов приёмника и передатчика. В этом случае приёмная и передающая сторона контроллера UART располагают лишь однобайтными буферными регистрами. Бит переполнения буфера устанавливается при этом тогда, когда очередной символ данных уже принят, однако предыдущий еще не был считан.</div> <div>В настоящей реализации модуля буферы FIFO физически не отключаются, необходимая функциональность достигается за счёт логических манипуляций с флагами. При этом в случае, если буфер FIFO отключен, а сдвиговый регистр передатчика пуст (не используется), запись байта данных происходит непосредственно в регистр сдвига, минуя буферный регистр.</div>			
Изм.	Лист	№ докум.	Подп.	Дата				
Изм.	Лист	№ докум.	Подп.	Дата				
Изм.	Лист	№ докум.	Подп.	Дата				
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.			Лист
Изм.	Лист	№ докум.	Подп.	Дата	Изм.</			

3.12.4.9 Проверка по шлейфу

Проверка по шлейфу (замыкание выхода передатчика на вход приёмника) выполняется путём установки в «1» бита LBE в регистре управления контроллером UARTCR.

3.12.4.10 Проверка по шлейфу

Проверка по шлейфу выполняется после установки в «1» бита LBE регистра управления контроллером UARTCR с одновременной установкой в «1» бита SIRSTEST регистра управления тестированием UARTTCCR.

В этом режиме данные, передаваемые на выход nSIROUT, должны подаваться на вход SIRIN.

Примечание — Это единственный случай использования тестового регистра в нормальном режиме функционирования модуля.

Временная диаграмма кадра передачи данных приведена на рисунке 3.12.3.

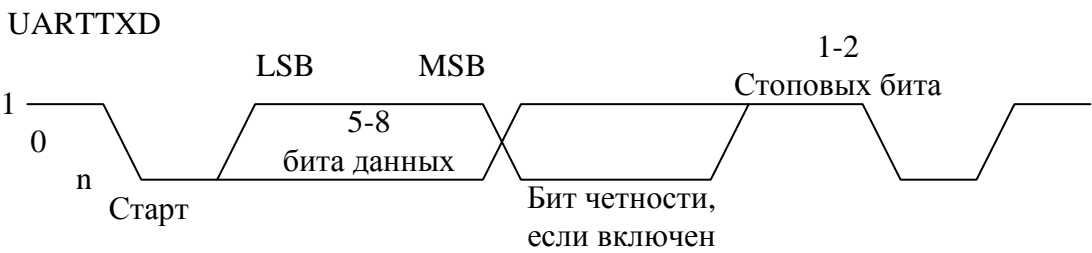


Рисунок 3.12.3 — Временная диаграмма кадра передачи данных

3.12.4.11 Интерфейс прямого доступа к памяти (DMA)

Модуль универсального асинхронного приёмопередатчика оснащён интерфейсом подключения к контроллеру прямого доступа к памяти. Функционирование в данном режиме контролируется регистром управления DMAUARTDMACR.

Интерфейс DMA включает в себя сигналы для приёма:

— UARTRXDMASREQ — запрос передачи отдельного символа, инициируется контроллером UART. Размер символа в режиме приёма данных — до 12 бит. Сигнал переводится в активное состояние в случае, если буфер FIFO приёмника содержит по меньшей мере один символ;

— UARTRXDMABREQ — запрос блочного обмена данными, инициируется модулем приёмопередатчика. Сигнал переходит в активное состояние в случае, если заполнение буфера FIFO приёмника превысило заданный порог. Порог программируется индивидуально для каждого буфера FIFO путём записи значения в регистр UARTIFLS;

— UARTRXDMACLR — сброс запроса на DMA, инициируется модулем приёмопередатчика с целью сброса принятого запроса. В случае если был запрошен блочный обмен данными, сигнал сброса формируется в ходе передачи последнего символа данных в блоке.

Интерфейс DMA включает в себя сигналы для передачи:

— UARTTXDMASREQ — запрос передачи отдельного символа, инициируется модулем приёмопередатчика. Размер символа в режиме передачи данных — до 8 бит. Сигнал переводится в активное состояние в случае, если буфер FIFO передатчика содержит, по меньшей мере, одну свободную ячейку;

— UARTTXDMABREQ — запрос блочного обмена данными, инициируется модулем приёмопередатчика. Сигнал переводится в активное состояние в случае, если заполнение буфера FIFO передатчика ниже заданного порога. Порог программируется индивидуально для каждого буфера FIFO путём записи значения в регистр UARTIFLS;

— UARTTXDMACLR — сброс запроса на DMA, инициируется контроллером DMA с целью сброса принятого запроса. В случае, если был запрошен блочный обмен данными, сигнал сброса формируется в ходе передачи последнего символа данных в блоке.

Сигналы блочного и одноэлементного обмена данными (UARTTXDMASREQ и UARTTXDMABREQ) не являются взаимно исключающими, они могут быть инициированы одновременно. Например, в случае, если заполнение данными буфера приёмника превышает пороговое значение, формируется как сигнал запроса одноэлементного обмена, так и сигнал запроса блочного обмена данными. В случае если количество данных в буфере приёма меньше порогового значения формируется

Име. № подл.	Подп. и дата	Взам. инв. №	Име. № дубл.	Подп. и дата.	Микросхема интегральная 5023BC016					Лист
					Техническое описание					
					Версия 2.04 от 01.09.2017 г.					171
					Изм.	Лист	№ докум.	Подп.	Дата	

только запрос одноэлементного обмена. Это бывает полезно в ситуациях, при которых объём данных меньше размера блока.

Пусть, например, нужно принять 19 символов, а порог заполнения буфера FIFO установлен равным четырём. Тогда контроллер DMA осуществит четыре передачи блоков по четыре символа, а оставшиеся три символа передаст в ходе трёх одноэлементных обменов.

Примечание — Для оставшихся трёх символов контроллер UART не может инициировать процедуру блочного обмена.

Каждый инициированный приёмопередатчиком сигнал запроса DMA остаётся активным до момента его сброса соответствующим сигналом DMACLR.

После снятия сигнала сброса модуль приёмопередатчика вновь получает возможность сформировать запрос на DMA в случае выполнения описанных выше условий. Все запросы DMA снимаются после запрета работы приёмопередатчика, а также в случае установки в ноль бита управления DMATXDMAE или RXDMAE в регистре управления DMAUARTDMACR.

В случае запрета буферов FIFO устройство способно передавать и принимать только одиночные символы; как следствие, контроллер может инициировать DMA только в одноэлементном режиме. При этом модуль в состоянии формировать только сигналы управления DMAUARTRXDMASREQ и UARTTXDMASREQ. Для информации о запрете буферов FIFO см. описание регистра управления линией UARTLCR_H.

Когда буферы FIFO включены, обмен данными может производиться в ходе как одноэлементных, так и блочных передач данных, в зависимости от установленной величины порога заполнения буферов и их фактического заполнения. Таблица 3.12.2 показывает значения параметров срабатывания запросов блочного обмена UARTRXDMABREQ и UARTTXDMABREQ в зависимости от порога заполнения буфера.

Ине. № подл.	Подп. и дата	Взам. инв. №	Ине. № дубл.	Подп. и дата.

					Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 172
Изм	Лист	№ докум.	Подп.	Дата		

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.

В регистре управления DMAUARTDMACR предусмотрен бит DMAONERR, который позволяет запретить DMA от приёмника в случае активного состояния линии прерывания по обнаружению ошибки UARTEINTR. При этом соответствующие линии запроса DMA — UARTRXDMAREQ и UARTRXDMAABREQ переводятся в неактивное состояние (маскируются) до сброса UARTEINTR. На линии запроса DMA, обслуживающие передатчик, состояние UARTEINTR не влияет.

The timing diagram illustrates the relationship between the PCLK clock and the DMA request and clear signals. The PCLK signal is a periodic square wave. The DMASREQ signal is active high and is asserted for a duration of 4 clock cycles. The DMABREQ signal is active high and is asserted for a duration of 4 clock cycles. The DMACLR signal is active low and is asserted for a duration of 4 clock cycles. The signals are shown on a grid with 10 vertical divisions and 4 horizontal divisions.

Необходимо иметь в виду, что при получении сигнала UARTTXDMABREQ контроллер DMA будет производить 2R_power транзакций независимо от уровня заполнения FIFO, поэтому при неверной настройке коэффициента R_power может

произойти переполнение FIFO и потеря данных. Размер транзакции DMA не должен быть более количества свободных ячеек в FIFO.

3.12.5 Прерывания

3.12.5.1 Общие положения

В модуле UART предусмотрено 9 маскируемых источников прерываний. В результате формируется один общий сигнал, представляющий собой комбинацию независимых сигналов, объединённых по схеме ИЛИ.

Сигналы запроса на прерывание:

- UARTRXINTR — прерывание от приёмника;
- UARTTXINTR — прерывание от передатчика;
- UARTRTINTR — прерывание по таймауту приёмника;
- UARTEINTR — ошибка;
- UARTOEINTR — переполнение буфера;
- UARTBEINTR — прерывание приёма — разрыв линии;
- UARTPEINTR — ошибка контроля чётности;
- UARTFEINTR — ошибка в структуре кадра.

UARTINTR — логическое ИЛИ сигналов UARTRXINTR, UARTTXINTR, UARTRTINTR и UARTEINTR. Каждый из независимых сигналов запроса на прерывание может быть маскирован путём установки соответствующего бита в регистре маски UARTIMSC. Установка бита в «1» разрешает соответствующее прерывание, в «0» — запрещает.

Доступность, как индивидуальных линий, так и общей линии запроса позволяет организовать обслуживание прерываний в системе, как путём применения глобальной процедуры обработки, так и с помощью драйвера устройства, построенного по модульному принципу.

Прерывания от приёмника и передатчика UARTRXINTR и UARTTXINTR выведены отдельно от прерываний по изменению состояния устройства. Это позволяет использовать сигналы запроса UARTRXINTR и UARTTXINTR для

Инв. № подл.	Подп. и дата	— UARTFEINTR — ошибка в структуре кадра.					
		UARTINTR — логическое ИЛИ сигналов UARTRXINTR, UARTTXINTR, UARTRTINTR и UARTEINTR. Каждый из независимых сигналов запроса на прерывание может быть маскирован путём установки соответствующего бита в регистре маски UARTIMSC. Установка бита в «1» разрешает соответствующее прерывание, в «0» — запрещает.					
		Доступность, как индивидуальных линий, так и общей линии запроса позволяет организовать обслуживание прерываний в системе, как путём применения глобальной процедуры обработки, так и с помощью драйвера устройства, построенного по модульному принципу.					
		Прерывания от приёмника и передатчика UARTRXINTR и UARTTXINTR выведены отдельно от прерываний по изменению состояния устройства. Это позволяет использовать сигналы запроса UARTRXINTR и UARTTXINTR для					
Изм.	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016		Лист
					Техническое описание		174
					Версия 2.04 от 01.09.2017 г.		

обеспечения чтения и записи данных согласованно с достижением заданного порога заполнения буферов FIFO приёмника и передатчика.

Признаки возникновения каждого из условий прерывания можно считать либо из регистра прерываний UARTRIS, либо из маскированного регистра прерываний UARTMIS.

3.12.5.2 UARTRXINTR

Состояние прерывания от приёмника может измениться в случае возникновения одного из следующих событий:

— буфер FIFO разрешён, и его заполнение достигло заданного порогового значения. В этом случае линия прерывания переходит в высокое состояние. Сигнал прерывания переходит в низкое состояние после чтения данных из буфера приёмника до тех пор, пока его заполнение не станет меньше порога, либо после сброса прерывания;

— буфер FIFO запрещён (имеет размер один символ), принят один символ данных. При этом линия прерывания переходит в высокое состояние. Сигнал прерывания переходит в низкое состояние после чтения одного байта данных, либо после сброса прерывания.

3.12.5.3 UARTTXINTR

Состояние прерывания от передатчика может измениться в случае возникновения одного из следующих событий:

— буфер FIFO разрешён и его заполнение меньше или равно заданному пороговому значению. В этом случае линия прерывания переходит в высокое состояние. Сигнал прерывания переходит в низкое состояние после записи данных в буфера передатчика до тех пор, пока его заполнение не станет больше порога, либо после сброса прерывания;

— буфер FIFO запрещён (имеет размер один символ), данные в буферном регистре передатчика отсутствуют. При этом линия прерывания переходит в высокое состояние.

Ине. № подл.	Подп. и дата	Взам. инв. №	Ине. № дубл.	Подп. и дата

					Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 175
Изм	Лист	№ докум.	Подп.	Дата		

Сигнал прерывания переходит в низкое состояние после записи одного байта данных, либо после сброса прерывания.

Для занесения данных в буфер FIFO передатчика необходимо записать данные в буфер либо перед разрешением работы приёмопередатчика и прерываний, либо после разрешения работы приёмопередатчика и прерываний.

Примечание — Прерывание передатчика срабатывает по фронту, а не по уровню сигнала.

В случае, если модуль и прерывания от него разрешены до осуществления записи данных в буфер FIFO передатчика, прерывание не формируется. Прерывание возникает только при опустошении буфера FIFO.

3.12.5.4 UARTRTINTR

Прерывание по таймауту приёмника возникает в случае, если буфер FIFO приёмника не пуст, и на вход приёмника не поступало новых данных в течение периода времени, необходимого для передачи 32 бит.

Прерывание по таймауту снимается либо после считывания данных из буфера приёмника до его опустошения (или считывания одного байта в случае, если буфер FIFO запрещён), либо путём записи «1» в соответствующий бит регистра сброса прерывания UARTICR.

3.12.5.5 UARTEINTR

Прерывание по обнаружению ошибки возникает в случае ошибки при приёме данных.

Оно может быть вызвано рядом факторов:

- ошибка в структуре кадра;
- ошибка контроля чётности;
- разрыв линии;
- переполнение буфера.

Причину возникновения прерывания можно определить, прочитав содержимое регистра прерываний UARTRIS, либо содержимое маскированного регистра прерываний UARTMIS.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	приёмника до его опустошения (или считывания одного байта в случае, если буфер FIFO запрещён), либо путём записи «1» в соответствующий бит регистра сброса прерывания UARTICR.								
					3.12.5.5 UARTEINTR								
					Прерывание по обнаружению ошибки возникает в случае ошибки при приёме данных.								
					Оно может быть вызвано рядом факторов:								
					— ошибка в структуре кадра;								
					— ошибка контроля чётности;								
					— разрыв линии;								
					— переполнение буфера.								
					Причину возникновения прерывания можно определить, прочитав содержимое регистра прерываний UARTRIS, либо содержимое маскированного регистра прерываний UARTMIS.								
										Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.		Лист	
												176	
Изм		Лист	№ докум.		Подп.	Дата							

Сброс прерывания осуществляется путём записи соответствующих бит в регистр сброса прерывания UARTICR. Прерываниям по обнаружению ошибки соответствуют биты с 7 по 10.

3.12.5.6 UARTINTR

Все описанные сигналы запроса на прерывание скомбинированы в общую линию путём объединения по схеме «ИЛИ» сигналов UARTRXINTR, UARTTXINTR, UARTRTINTR и UARTEINTR с учётом маскирования. Общий выход может быть подключен к системному контроллеру прерывания, что позволит ввести дополнительное маскирование запросов на уровне периферийных устройств.

3.12.6 Программное управление модулем

3.12.6.1 Общая информация, применимая ко всем регистрам контроллера

Базовый адрес контроллера не фиксирован и может быть различным в разных системах. Смещение каждого регистра относительно базового адреса постоянно.

Запрещён доступ к зарезервированным или неиспользуемым адресам. Это может привести к непредсказуемому поведению модуля, за исключением специально оговоренных в настоящем документе случаев:

— запрещено изменение значений не определённых в документе разрядов регистров;

— все биты регистров (за исключением специально оговоренных случаев) устанавливаются в значение «0» после сброса по включению питания или системного сброса.

Столбец «Тип» таблицы 3.12.3 определяет режим доступа к регистру в соответствии с обозначениями:

— RW — чтение и запись;

— RO — только чтение;

— WO — только запись.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>					Лист
Изм.	Лист	№ докум.	Подп.	Дата						177

3.12.6.2 Обобщённые данные о регистрах устройства

Данные о регистрах модуля универсального асинхронного приёмопередатчика приведены в таблице 3.12.3.

Таблица 3.12.3 — Описание регистров модуля универсального асинхронного приёмопередатчика UARTx

Смещение	Название регистра	Тип	Значение по сбросу	Ширина	Описание
0x000	DR	RW	0x	12/8	UARTx -> DR Регистр данных
0x004	RSR_ECR	RW	0x0	4/0	UARTx -> RSR_ECR Регистр состояния приёмника / Сброс ошибки приёмника
0x008-0x014					Зарезервировано
0x018	FR	RO	0b-10010	9	UARTx -> FR Регистр флагов
0x01C					Зарезервировано
0x020	—				Зарезервировано
0x024	IBRD	RW	0x0000	16	UARTx -> IBRD Целая часть делителя скорости обмена данными
0x028	FBRD	RW	0x00	6	UARTx -> FBRD Дробная часть делителя скорости обмена данными
0x02C	LCR_H	RW	0x00	8	UARTx -> LCR_H Регистр управления линией
0x030	CR	RW	0x0300	16	UARTx -> CR Регистр управления
0x034	IFLS	RW	0x12	6	UARTx -> IFLS Регистр порога прерывания по заполнению буфера FIFO
0x038	FMSC	RW	0x000	11	UARTx -> IMSC Регистр маски прерывания
0x03C	RIS	RO	0x00-	11	UARTx -> RIS Регистр состояния прерываний
0x040	MIS	RO	0x00-	11	UARTx -> MIS Регистр состояния прерываний с маскированием
0x044	ICR	WO	—	11	UARTx -> ICR Регистр сброса прерывания
0x048	DMACR	RW	0x00	3	UARTx -> DMACR Регистр управления DMA

Ине. № подл.	Подп. и дата.
Взам. инв. №	Ине. № дубл.
Подп. и дата	
Ине. № подл.	

Изм	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 178
-----	------	----------	-------	------	---	-------------

3.12.6.3 Регистр UARTx -> DR (регистр данных)

В ходе передачи данных, если буфер FIFO передатчика разрешён, то слово данных, записанное в рассматриваемый регистр, направляется в буфер FIFO передатчика.

В противном случае, записанное слово фиксируется в буферный регистр передатчика (последний элемент буфера FIFO).

Операция записи в регистр инициирует передачу данных. Слово данных предваряется стартовым битом, дополняется битом контроля чётности (если режим контроля четности включен) и стоповым битом. Сформированное слово отправляется в линию передачи данных.

В ходе приёма данных, если буфер FIFO приёмника разрешён, байт данных и четыре бита состояния (разрыв, ошибка формирования кадра, чётность, переполнение) сохраняются в 12-битном буфере.

В противном случае байт данных и биты состояния записываются в буферный регистр (последний элемент буфера FIFO).

Полученные из линии связи байты данных считывается путём чтения из регистра DR принятых данных совместно с соответствующими битами состояния. Информация о состоянии также может быть получена путём чтения регистра UARTRSR/UARTECR (таблица 3.12.4).

Таблица 3.12.4 — Описание регистра UARTx->DR

№ бита	Наименование бита	Функциональное назначение бита, краткое описание назначения и принимаемых значений
1	2	3
15 – 12		Резерв
11	OE	Переполнение буфера приёмника. Бит устанавливается в «1» в случае, если на вход приёмника поступают данные, когда буфер заполнен. Сбрасывается в «0» после того, как в буфере появится свободное место
10	BE	Разрыв линии. Устанавливается в «1» при обнаружении признака разрыва линии, т.е. в случае наличия низкого логического уровня на входе приёмника в течение времени, большего чем длительность передачи полного слова данных (включая стартовый, стоповый биты и бит проверки на чётность). При включенном FIFO данная ошибка ассоциируется с последним символом, поступившим в буфер. В случае обнаружения разрыва линии в буфер загружается только один нулевой символ, приём данных возобновляется только после перехода линии в логическую «1» и последующего обнаружения корректного стартового бита

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	<div>Микросхема интегральная 5023BC016</div> <div>Техническое описание</div> <div>Версия 2.04 от 01.09.2017 г.</div> <div>Лист 179</div>				
Изм.	Лист	№ докум.	Подп.	Дата					

Продолжение таблицы 3.12.4

1	2	3
9	PE	Ошибка контроля чётности. Устанавливается в «1» в случае, если чётность принятого символа данных не соответствует установкам бит EPS и SPS в регистре управления линией UARTLCR_H. При включенном FIFO данная ошибка ассоциируется с последним символом, поступившим в буфер
8	FE	Ошибка в структуре кадра. Устанавливается в «1» в случае, если в принятом символе не обнаружен корректный стоповый бит (корректный стоповый бит равен 1). При включенном FIFO данная ошибка ассоциируется с последним символом, поступившим в буфер
7 – 0	DATA	Принимаемые данные (чтение). Передаваемые данные (запись)

Примечание — Необходимо запрещать работу приёмопередатчика перед любым перепрограммированием его регистров управления. Если приёмопередатчик переводится в отключенное состояние во время передачи или приёма символа, то перед остановкой он завершает выполняемую операцию.

3.12.6.4 Регистр UARTx -> RSR_ECR (регистр состояния приёмника/сброса ошибки)

Состояние приёмника также может быть считано из регистра UARTx -> RSR_ECR. В этом случае информация о состоянии признаков разрыва линии, ошибки контроля чётности и ошибки в структуре кадра относится к последнему символу, считанному из регистра данных DR.

С другой стороны, признак переполнения буфера устанавливается немедленно после возникновения этого состояния (и не связан с последним считанным из регистра DR байтом данных).

Запись в регистр UARTx -> RSR_ECR приводит к сбросу признаков ошибок переполнения, чётности, структуры кадра, разрыва линии. Кроме того, все эти признаки устанавливаются в «0» после сброса устройства.

Таблица 3.12.5 содержит описание регистров UARTx -> RSR_ECR.

Изм.	Лист	№ докум.	Подп.	Дата	Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	<p>Состояние приёмника также может быть считано из регистра UARTx -> RSR_ECR. В этом случае информация о состоянии признаков разрыва линии, ошибки контроля чётности и ошибки в структуре кадра относится к последнему символу, считанному из регистра данных DR.</p> <p>С другой стороны, признак переполнения буфера устанавливается немедленно после возникновения этого состояния (и не связан с последним считанным из регистра DR байтом данных).</p> <p>Запись в регистр UARTx -> RSR_ECR приводит к сбросу признаков ошибок переполнения, чётности, структуры кадра, разрыва линии. Кроме того, все эти признаки устанавливаются в «0» после сброса устройства.</p> <p>Таблица 3.12.5 содержит описание регистров UARTx -> RSR_ECR.</p>
<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>										Лист
										180

Таблица 3.12.5 — Описание регистров UARTx -> RSR_ECR

№ бита	Наименование бита	Функциональное назначение бита, краткое описание назначения и принимаемых значений.
7 – 4		Резерв, при чтении результат не определен.
3	OE	Переполнение буфера приёмника. Бит устанавливается в «1» в случае, если на вход приёмника поступают данные, в то время как буфер заполнен. Сбрасывается в «0» после записи в регистр UARTECR. Содержимое буфера остается верным, так как перезаписан был только регистр сдвига. Центральный процессор должен считать данные для того, чтобы освободить буфер FIFO.
2	BE	Разрыв линии. Устанавливается в «1» при обнаружении признака разрыва линии, то есть в случае наличия низкого логического уровня на входе приёмника в течение времени, большего чем длительность передачи полного слова данных (включая стартовый, стоповый биты и бит проверки на четность). Бит сбрасывается в «0» после записи в регистр UARTECR. При включенном FIFO данная ошибка ассоциируется с символом, находящемся на вершине буфера. В случае обнаружения разрыва линии в буфер загружается только один нулевой символ, приём данных возобновляется только после перехода линии в логическую 1 и последующего обнаружения корректного стартового бита
1	PE	Ошибка контроля чётности. Устанавливается в «1» в случае, если чётность принятого символа данных не соответствует установкам бит EPS и SPS в регистре управления линией UARTECR_N (стр. 3 – 12). Бит сбрасывается в «0» после записи в регистр UARTECR. При включенном FIFO данная ошибка ассоциируется с символом, находящимся на вершине буфера.
0	FE	Ошибка в структуре кадра. Устанавливается в «1» в случае, если в принятом символе не обнаружен корректный стоповый бит (корректный стоповый бит равен 1). Бит сбрасывается в «0» после записи в регистр UARTECR. При включенном FIFO данная ошибка ассоциируется с символом, находящимся на вершине буфера.

Примечания:

1. Перед чтением регистра состояния UARTRSR необходимо считать данные, принятые из линии, путём обращения к регистру данных DR. Противоположная последовательность действий не допускается, так как регистр UARTRSR обновляет свое состояние только после чтения регистра DR. Вместе с тем, информация о состоянии приёмника может быть получена непосредственно из регистра данных DR.

2. Запись в регистр UARTRSR/UARTECR любого кода сбрасывает признаки ошибок формирования кадра, проверки на четность, разрыва линии и переполнения буфера.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата					
Изм	Лист	№ докум.	Подп.	Дата					
					Микросхема интегральная 5023BC016				
					Техническое описание				
					Версия 2.04 от 01.09.2017 г.				
					Лист				
					181				

3.12.6.5 Регистр UARTx -> FR (регистр флагов)

После сброса биты регистра флагов TXFF, RXFF и BUSY устанавливаются в 0, а биты TXFE и RXFE — в 1. В таблице 3.12.6 представлено описание регистра UARTFR.

Таблица 3.12.6 — Описание регистра UARTx -> FR

№ бита	Наименование бита	Функциональное назначение бита, краткое описание назначения и принимаемых значений
15 – 9	–	Резерв. Не модифицируется. При чтении заполняются нулями
8	RI	Инверсия линии nUARTRI
7	TXFE	Буфер FIFO передатчика пуст. Значение бита зависит от состояния бита FEN регистра управления линией UARTLCR_H. Если буфер FIFO запрещён, бит устанавливается в «1» когда буферный регистр передатчика пуст. В противном случае он равен 1 если пуст буфер FIFO передатчика. Данный бит не даёт никакой информации о наличии данных в регистре сдвига передатчика
6	RXFF	Буфер FIFO приёмника заполнен. Значение бита зависит от состояния бита FEN регистра управления линией UARTLCR_H. Если буфер FIFO запрещён, бит устанавливается в «1» когда буферный регистр приёмника занят. В противном случае он равен 1 если заполнен буфер FIFO приёмника
5	TXFF	Буфер FIFO передатчика заполнен. Значение бита зависит от состояния бита FEN регистра управления линией UARTLCR_H. Если буфер FIFO запрещён, бит равен 1 когда буферный регистр передатчика занят. В противном случае он равен 1 если заполнен буфер FIFO передатчика
4	RXFE	Буфер FIFO приёмника пуст. Значение бита зависит от состояния бита FEN регистра управления линией UARTLCR_H. Если буфер FIFO запрещён, бит устанавливается в «1» когда буферный регистр приёмника пуст. В противном случае он равен 1 если пуст буфер FIFO приёмника
3	BUSY	UART занят. Бит равен 1 в случае, если контроллер передает в линию данные. Бит остается установленным до тех пор, пока данные, включая стоповые биты, не будут полностью переданы. Кроме того, бит занятости устанавливается в «1» при наличии данных в буфере FIFO передатчика, вне зависимости от состояния приёмопередатчика (даже если он запрещён)
2 – 0	–	Не используются

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.

Изм	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 182
-----	------	----------	-------	------	---	-------------

Продолжение таблицы 3.12.9

1	2	3
4	FEN	Разрешение работы буфера FIFO приёмника и передатчика. 0 — запрещено; 1 — разрешено.
3	STP2	Режим передачи двух стоповых бит. 0 — один стоповый бит; 1 — два стоповых бита. Приёмник не проверяет наличие дополнительного стопового бита в кадре.
2	EPS	Чётность/нечётность. 0 — бит чётности дополняет количество единиц в информационной части кадра до нечётного; 1 — до чётного числа. Значение бита EPS не играет роли в случае, если битом PEN формирование и проверка бита чётности запрещена.
1	PEN	Разрешение проверки чётности. 0 — кадр не содержит бита чётности; 1 — бит чётности передается в кадре и проверяется при приёме данных.
0	BRK	Разрыв линии. Если этот бит установлен в «1», то по завершении передачи текущего символа на выходе UARTTXD устанавливается низкий уровень сигнала. Для правильного выполнения этой операции программное обеспечение должно обеспечить передачу сигнала разрыва в течение, как минимум, времени передачи двух информационных кадров. В нормальном режиме функционирования бит должен быть установлен в «0».

Содержимое регистров UARTLCR_H, UARTIBRD и UARTFBRD совместно образует общий 30-разрядный регистр UARTLCR, который обновляется по стробу, формируемому при записи в UARTLCR_H. Таким образом, для того, чтобы изменение параметров коэффициента деления частоты обмена данными вступило в силу, после изменения значения регистров UARTIBRD и/или UARTFBRD необходимо осуществить запись данных в регистр UARTLCR_H.

Примечание — Изменение значений трёх регистров можно осуществить корректно двумя способами:

- запись UARTIBRD, запись UARTFBRD, запись UARTLCR_H;
- запись UARTFBRD, запись UARTIBRD, запись UARTLCR_H.

Для того, чтобы изменить значение лишь одного из регистров (UARTIBRD или UARTFBRD), необходимо выполнить запись UARTIBRD (или UARTFBRD), запись UARTLCR_H.

Име. № подл.	Подп. и дата.	Взам. инв. №	Инв. № дубл.	Подп. и дата.	<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>					Лист
										185
Изм.	Лист	№ докум.	Подп.	Дата						

В таблице 3.12.10 представлена таблица истинности для бит управления контролем чётности PEN, EPS и SPS регистра управления линией UARTLCR_H.

Таблица 3.12.10 — Управление режимом контроля четности

PEN	EPS	SPS	Бит контроля чётности
0	X	X	Не передается, не проверяется
1	1	0	Проверка чётности слова данных
1	0	0	Проверка нечётности слова данных
1	0	1	Бит чётности постоянно равен 1
1	1	1	Бит чётности постоянно равен 0

Примечания:

1) Регистры UARTLCR_H, UARTIBRD и UARTFBRD не должны изменяться при следующих условиях:

- при разрешённой работе приёмопередатчика;
- во время завершения приёма или передачи данных в процессе остановки (перевода в запрещённое состояние) приёмопередатчика;

2) Целостность данных в буферах FIFO не гарантируется в следующих случаях:

- после установки бита разрыва линии BRK;
- если программное обеспечение произвело остановку приёмопередатчика при наличии данных в буферах FIFO после его повторного перевода в разрешённое состояние.

3.12.6.9 Регистр UARTx -> CR (регистр управления)

После сброса все биты регистра управления UARTx -> CR, за исключением бит 9 и 8 устанавливаются в нулевое состояние. Биты 9 и 8 устанавливаются в единичное состояние.

Назначение разрядов регистра управления приведено в таблице 3.12.11.

Таблица 3.12.11 — Описание регистра управления UARTx -> CR

№ бита	Наименование бита	Функциональное назначение бита, краткое описание назначения и принимаемых значений
1	2	3
15 – 10	–	

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>					Лист
Изм.	Лист	№ докум.	Подп.	Дата						186

Продолжение таблицы 3.12.11

1	2	3
9	RXE	Приём разрешён. Установка бита в «1» разрешает работу приёмника. Приём данных осуществляется по интерфейсу асинхронного последовательного обмена В случае перевода приёмопередатчика в запрещённое состояние в ходе приёма данных, он завершает приём текущего символа перед остановкой.
8	TXE	Передача разрешена. Установка бита в «1» разрешает работу передатчика. Передача осуществляется по интерфейсу асинхронного последовательного обмена. В случае перевода приёмопередатчик в запрещённое состояние в ходе передачи данных, он завершает передачу текущего символа: перед остановкой.
7	LBE	0 — запрещено; 1 — шлейф разрешён. выходная линия передатчика UARTTXD коммутируется на вход приёмника UARTRXD. После сброса бит устанавливается в «0»
6 – 3		Резерв. Не модифицируется. При чтении выдаются нули.
2	–	Не используется. Должны быть нули.
1	–	Не используется. Должны быть нули.
0	UARTEN	Разрешение работы приёмопередатчика: 0 — функционирование запрещено. Перед остановкой завершается приём и/или передача обрабатываемого в текущий момент символа. 1 — функционирование разрешено. Производится обмен данными либо по линиям асинхронного обмена.

Примечания:

1) Для того, чтобы разрешить передачу данных, необходимо установить в логическую 1 биты TXE и UARTEN. Аналогично, для разрешения приёма данных необходимо установить в 1 биты RXE и UARTEN.

2) Рекомендуются следующая последовательность действий для программирования регистров управления:

- остановить работу приёмопередатчика;
- дождаться окончания приёма и/или передачи текущего символа данных;
- сброс буфера передатчика путём установки бита FEN регистра UARTLCR_N в 0;
- изменение настроек регистра UARTCR;
- возобновление работы приёмопередатчика.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.
Изм	Лист	№ докум.	Подп.	Дата
Микросхема интегральная 5023BC016				Лист
Техническое описание				187
Версия 2.04 от 01.09.2017 г.				

3.12.6.10 Регистр UARTx -> IFLS (регистр порога прерывания по заполнению буфера FIFO)

Регистр UARTx -> IFLS используется для установки порогового значения заполнения буферов передатчика и приёмника, по достижению которых генерируется сигнал прерывания UARTRXINTR или UARTTXINTR, соответственно. Прерывание генерируется в момент перехода величины заполнения буфера через заданное значение.

После сброса в регистре устанавливается порог, соответствующий заполнению половины буфера. Описание регистра UARTx -> IFLS и значения представлен в таблице 3.12.12.

Таблица 3.12.12 — Описание регистра UARTx -> IFLS

№ бита	Наименование бита	Функциональное назначение бита, краткое описание назначения и принимаемых значений
31 – 6	–	
5 – 3	RXIFLSEL	Порог прерывания по заполнению буфера приёмника: b000 = буфер заполнен на 1/8; b001 = буфер заполнен на 1/4; b010 = буфер заполнен на 1/2; b011 = буфер заполнен на 3/4; b100 = буфер заполнен на 7/8; b101 – b111 = резерв.
2 – 0	TXIFLSEL	Порог прерывания по заполнению буфера передатчика: b000 = буфер заполнен на 1/8; b001 = буфер заполнен на 1/4; b010 = буфер заполнен на 1/2; b011 = буфер заполнен на 3/4; b100 = буфер заполнен на 7/8; b101 – b111 = резерв.

3.12.6.11 Регистр UARTx -> IMSC (регистр установки сброса маски прерывания)

При чтении регистра UARTx -> IMSC выдается текущее значение маски. При записи производится установка или сброс маски на соответствующее прерывание.

После сброса все биты регистра маски устанавливаются в нулевое состояние. Назначение бит регистра UARTx -> IMSC содержится в таблице 3.12.13.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	<div>Микросхема интегральная 5023BC016</div> <div>Техническое описание</div> <div>Версия 2.04 от 01.09.2017 г.</div> <div>Лист 188</div>				
Изм	Лист	№ докум.	Подп.	Дата					

Таблица 3.12.13 — Описание регистра UARTx -> IMSC

№ бита	Наименование бита	Функциональное назначение бита, краткое описание назначения и принимаемых значений.
31 – 11	–	
10	OEIM	Маска прерывания по переполнению буфера UARTOEINTR: 1 — установлена; 0 — сброшена.
9	BEIM	Маска прерывания по разрыву линии UARTBEINTR: 1 — установлена; 0 — сброшена.
8	PEIM	Маска прерывания по ошибке контроля чётности UARTPEINTR: 1 — установлена; 0 — сброшена.
7	FEIM	Маска прерывания по ошибке в структуре кадра UARTFEINTR: 1 — установлена; 0 — сброшена.
6	RTIM	Маска прерывания по таймауту приёма данных UARTRTINTR: 1 — установлена; 0 — сброшена.
5	TXIM	Маска прерывания от передатчика UARTTXINTR. 1 — установлена; 0 — сброшена.
4	RXIM	Маска прерывания от приёмника UARTRXINTR. 1 — установлена; 0 — сброшена.
3 – 0	–	Не используется.

3.12.6.12 Регистр UARTx -> RIS (регистр состояния прерываний)

Регистр UARTx -> RIS доступен только для чтения и содержит текущее состояние прерываний без учета маскирования. Данные, записываемые в регистр UARTx -> RIS, игнорируются.

Предупреждение — После сброса все биты регистра, за исключением бит прерывания по состоянию модема (биты с 3 по 0), устанавливаются в «0». Значение бит прерывания по состоянию модема после сброса не определено.

Описание регистра UARTx -> RIS представлено в таблице 3.12.14.

Таблица 3.12.14 — Описание регистра UARTx -> RIS

№ бита	Наименование бита	Функциональное назначение бита, краткое описание назначения и принимаемых значений
1	2	3
31 – 11	–	
10	OERIS	Состояние прерывания по переполнению буфера UARTOEINTR.
9	BERIS	Состояние прерывания по разрыву линии UARTBEINTR.
8	PERIS	Состояние прерывания по ошибке контроля чётности UARTPEINTR.
7	FERIS	Состояние прерывания по ошибке в структуре кадра UARTFEINTR.
6	RTRIS	Состояние прерывания по таймауту приёма данных UARTRTINTR.

Име. № подл.	Подп. и дата	Взам. име. №	Име. № дубл.	Подп. и дата.

Изм	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 189
-----	------	----------	-------	------	---	-------------

Продолжение таблицы 3.12.4

1	2	3
5	TXRIS	Состояние прерывания от передатчика UARTTXINTR.
4	RXRIS	Состояние прерывания от приёмника UARTRXINTR.
3 – 0	–	Не используется

3.12.6.13 Регистр UARTx -> MIS (регистр маскированного состояния прерываний)

Регистр UARTx -> MIS доступен только для чтения и содержит текущее состояние прерываний с учетом маскирования. Данные, записываемые в регистр, игнорируются.

После сброса все биты регистра, за исключением бит прерывания по состоянию модема (биты с 3 по 0), устанавливаются в «0». Значение бит прерывания по состоянию модема после сброса не определено.

Описание регистра UARTx -> MIS представлено в таблице 3.12.15.

Таблица 3.12.15 — Описание регистра UARTx -> MIS

№ бита	Наименование бита	Функциональное назначение бита, краткое описание назначения и принимаемых значений.
31 – 11	–	
10	OEMIS	Маскированное состояние прерывания по переполнению буфера UARTOEINTR.
9	BEMIS	Маскированное состояние прерывания по разрыву линии UARTBEINTR.
8	PEMIS	Маскированное состояние прерывания по ошибке контроля чётности UARTPEINTR.
7	FEMIS	Маскированное состояние прерывания по ошибке в структуре кадра UARTFEINTR.
6	RTMIS	Маскированное состояние прерывания по таймауту приёма данных UARTRTINTR.
5	TXMIS	Маскированное состояние прерывания от передатчика UARTTXINTR.
4	RXMIS	Маскированное состояние прерывания от приёмника UARTRXINTR.
3 – 0	0	Не используется. Должны быть нули.

3.12.6.14 Регистр UARTx -> ICR (регистр сброса прерываний)

Регистр UARTx -> ICR доступен только для записи и предназначен для сброса признака прерывания по заданному событию путём записи «1» в соответствующий бит. Запись нуля в любой из разрядов регистра игнорируется.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.					Лист
					Изм	Лист	№ докум.	Подп.	Дата	
										190

Описание регистра UARTx -> ICR представлено в таблице 3.12.16.

Таблица 3.12.16 — Описание регистра UARTx -> ICR

№ бита	Наименование бита	Функциональное назначение бита, краткое описание назначения и принимаемых значений
31 – 11	–	
10	OEIC	Сброс прерывания по переполнению буфера UARTOEINTR.
9	BEIC	Сброс прерывания по разрыву линии UARTBEINTR.
8	PEIC	Сброс прерывания по ошибке контроля чётности UARTPEINTR.
7	FEIC	Сброс прерывания по ошибке в структуре кадра UARTFEINTR.
6	RTIC	Сброс прерывания по таймауту приёма данных UARTRTINTR.
5	TXIC	Сброс прерывания от передатчика UARTTXINTR.
4	RXIC	Сброс прерывания от приёмника UARTRXINTR.
3 – 0	–	Не используется. Должны быть нули.

3.12.6.15 Регистр UARTx -> DMACR (регистр управления прямым доступом к памяти)

Регистр UARTx -> DMACR доступен по чтению и записи. После сброса все биты регистра обнуляются.

Описание регистра UARTx -> DMACR представлено в таблице 3.12.17.

Таблица 3.12.17 — Описание регистра UARTx -> DMACR

№ бита	Наименование бита	Функциональное назначение бита, краткое описание назначения и принимаемых значений
31 – 3	–	
2	DMAONERR	Если бит установлен в «1», то в случае возникновения прерывания по обнаружению ошибки блокируются запросы DMA от приёмника UARTRXDMASREQ и UARTRXDMABREQ.
1	TXDMAE	Использование DMA при передаче. Если бит установлен в «1», то разрешено формирование запросов DMA для обслуживания буфера FIFO передатчика.
0	RXDMAE	Использование DMA при приёме. Если бит установлен в «1», то разрешено формирование запросов DMA для обслуживания буфера FIFO приёмника.

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	Микросхема интегральная 5023BC016					Лист
					Техническое описание					
					Версия 2.04 от 01.09.2017 г.					191
					Изм.	Лист	№ докум.	Подп.	Дата	

3.13 Таймер общего назначения

3.13.1 Общее описание

Таймер общего назначения представляет собой 32-битный счётчик с обратным отсчётом.

Когда счётчик доходит до 0, происходит генерация сигнала прерывания TIMERINT. Сигнал прерывания удерживается в активном состоянии до сброса регистра статуса прерывания INTCLEAR.

Если счётчик дошёл до 0, и в тот же момент программа очистила предыдущий статус прерывания, то статус прерывания устанавливается в «1».

Структурная схема таймера представлена на рисунке 3.13.1.

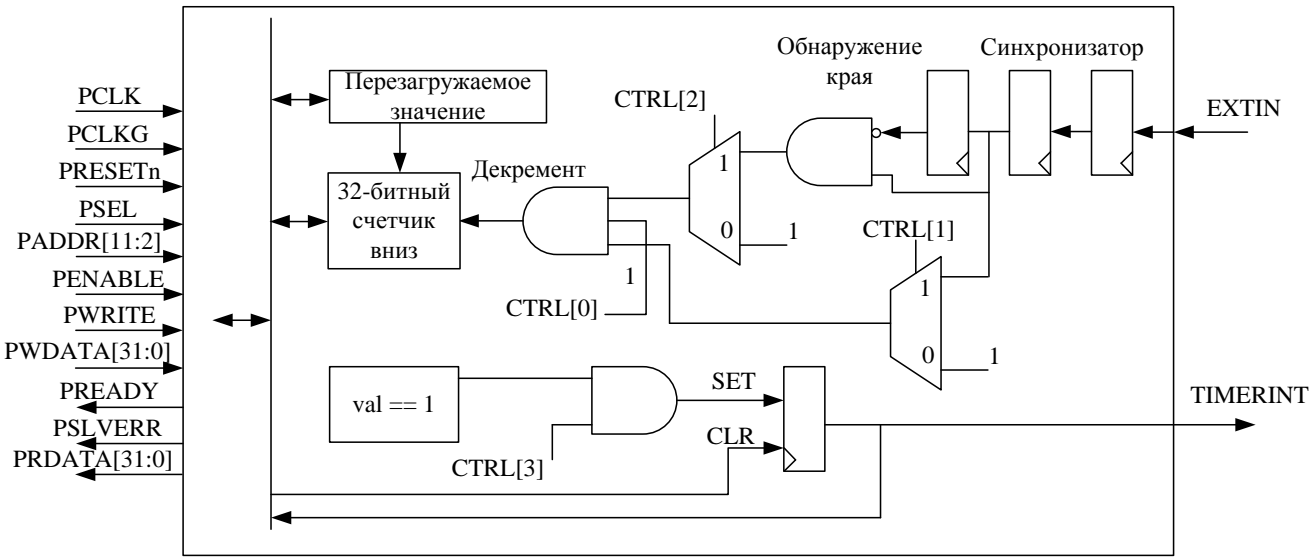


Рисунок 3.13.1 — Структурная схема таймера APB

В таблице 3.13.1 приведено распределение адресного пространства таймера APB

Таблица 3.13.1 — Регистры интерфейса таймера

Смещение	Наименование регистра	Тип	Значение по сбросу	Описание
0x000	CTRL	RW	0x0	[3] Разрешение прерывания таймера; [2] Всегда «0»; [1] Всегда «0»; [0] Включение.
0x004	VALUE	RW	0x00000000	[31:0] Текущее значение
0x008	RELOAD	RW	0x00000000	[31:0] Значение перезапуска. Запись в этот регистр задает текущее значение.
0x00C	INTCLEAR	RW	0x00000000	Регистр сброса прерывания

3.14 Сторожевой таймер

3.14.1 Описание сторожевого таймера

Сторожевой таймер позволяет сбросить систему в случае отказа программного обеспечения. Пользователь может включать или выключать таймер по собственному усмотрению.

Сторожевой таймер представляет собой 32-битный счётчик с обратным отсчётом, который инициализируется из регистра WDOGLOAD. Счётчик может быть сброшен любой записью в регистр WDOGINTCCLR. Когда счётчик достигает значение «0» без сброса его записью в регистр WDOGINTCCLR генерируется прерывание и счётчик устанавливается в значение WDOGLOAD. Далее продолжается обратный отсчёт. Если счётчик повторно достигает значения «0» без сброса любой записью в регистр WDOGINTCCLR, то сторожевой таймер устанавливает сигнал WDOGRES сброса процессора.

На рисунке 3.14.1 представлена диаграмма состояний сторожевого таймера.

3.14.2 Регистры сторожевого таймера (Watchdog)

Формат регистров сторожевого таймера (Watchdog) приведён в таблице 3.14.1.

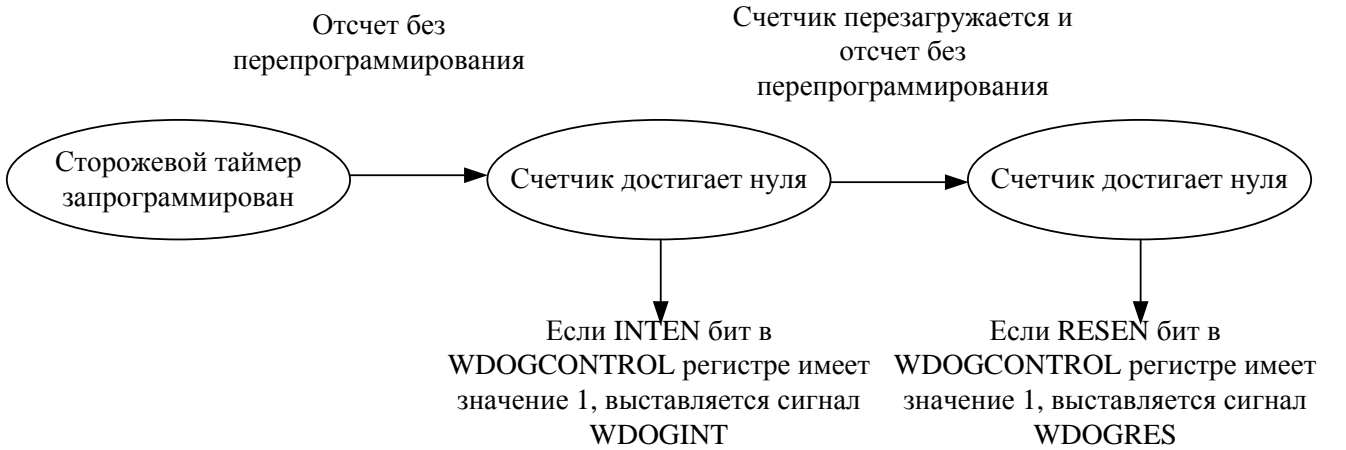


Рисунок 3.14.1 — Диаграмма состояний сторожевого таймера

Ине. № подл.	Подп. и дата.
Взам. инв. №	Инв. № дубл.
Подп. и дата	
Изм	Лист
№ докум.	Подп.
Дата	

Таблица 3.14.1 — Формат регистров сторожевого таймера

Смещение	Наименование регистра	Тип	Значение по сбросу	Описание
0x00	WDOGLOAD	RW	0xFFFFFFFF	Описание регистров содержится в отдельных таблицах ниже.
0x04	WDOGVALUE	RO	0xFFFFFFFF	
0x08	WDOGCONTROL	RW	0x0	
0x0C	WDOGINTCLR	WO	—	
0x10	WDOGRIS	RO	0x0	
0x14	WDOGMIS	RO	0x0	
0xC00	WDOGLOCK	RW	0x0	
0xF00	WDOGITCR	RW	0x0	
0xF04	WDOGITOP	WO	0x0	

3.14.2.1 WDOGLOAD (Watchdog Load Register)

32-битный регистр WDOGLOAD, хранит начальное значение счётчика. Когда происходит запись в этот регистр, счётчик сразу иницируется новым значением. Минимальное допустимое значение счёта составляет 0x00000001.

3.14.2.2 WDOGVALUE (Watchdog Value Register)

WDOGVALUE — текущее значение счётчика.

3.14.2.3 WDOGCONTROL (Watchdog Control Register)

WDOGCONTROL — регистр программного управления сторожевым таймером, доступный для чтения и записи. Описание регистра WDOGCONTROL приведено в таблице 3.14.2.

Таблица 3.14.2 — Описание регистра WDOGCONTROL

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 2	—	
1	RESEN	Разрешает выход сброса сторожевого таймера WDOGRES. Работает как маска по функции «Логическое И» для этого выхода. Устанавливается в «1» для включения сброса, в «0» — для выключения.
0	INTEN	Разрешает выход прерывания WDOGINT. Устанавливается в «1» для включения счётчика и срабатывания прерывания и в «0» — для выключения счётчика и выключения прерывания. Если счётчик был выключен, то после включения он иницируется значением из регистра WDOGLOAD.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.					
Изм	Лист	№ докум.	Подп.	Дата					
					Микросхема интегральная 5023BC016				
					Техническое описание				
					Версия 2.04 от 01.09.2017 г.				
					Лист				
					194				

3.14.2.4 WDOGINTCLR (Watchdog Clear Interrupt Register)

Запись любого значения в регистр WDOGINTCLR приводит к сбросу прерывания сторожевого таймера и сброса счётчика в значение из регистра WDOGLOAD.

3.14.2.5 WDOGRIS (Watchdog Raw Interrupt Status Register)

Регистр WDOGRIS является регистром первичного прерывания (rawinterrupt), доступным только для чтения. Над значением, хранящемся в регистре, производится вычисление функции и с битом разрешения прерываний INTEN.

Описание регистра WDOGRIS приведено в таблице 3.14.3.

Таблица 3.14.3 — Описание регистра WDOGRIS

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 1	–	Зарезервировано
0	Raw Watchdog Interrupt	Значение немаскированного бита прерывания

3.14.2.6 WDOGMIS (Watchdog Interrupt Status Register)

Регистр прерывания WDOGMIS доступен только для чтения. Регистр сигнализирует о появлении маскированного прерывания из счётчика. Это значение получается путём вычисления логического «И» со значениями первичного прерывания (rawinterrupt) из регистра WDORIS и бита включения прерывания INTEN из регистра управления WDOGCONTROL.

Описание регистра WDOGMIS приведено в таблице 3.14.4.

Таблица 3.14.4 — Описание регистра WDOGMIS

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 1	–	Зарезервировано
0	Watchdog Interrupt	Значение маскированного бита прерывания

3.14.2.7 WDOGLOCK (Watchdog Lock Register)

Регистр WDOGLOCK доступен только для записи. Использование этого регистра позволяет запретить запись во все остальные регистры сторожевого таймера. Это необходимо для запрета ненадежным программам отключать функции

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	<div>Микросхема интегральная 5023BC016</div> <div>Техническое описание</div> <div>Версия 2.04 от 01.09.2017 г.</div> <div>Лист 195</div>				
Изм.	Лист	№ докум.	Подп.	Дата					

сторожевого таймера. Запись значения 0x1ACCE551 разрешает запись во все остальные регистры, запись любого другого отключает возможность записи в другие регистры. Чтение этого регистра возвращает только 1 бит (нулевой).

Описание регистра WDOGLOCK приведено в таблице 3.14.5.

Таблица 3.14.5 — Описание регистра WDOGLOCK

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 1	–	Зарезервировано
0	Register Write enable status	0 — Запись во все регистры разрешена, значение по умолчанию. 1 — запись во все регистры запрещена.

3.14.2.8 WDOGITCR

Регистр Watchdog Integration Test Control Register (WDOGITCR) поддерживает запись и чтение. Это одноканальный регистр, который включает модуль интеграционного тестирования.

В режиме интеграционного тестирования порты WDOGINТ и WDOGRES управляются из регистра WDOGITOP.

Описание регистра WDOGITCR приведено в таблице 3.14.6.

Таблица 3.14.6 — Описание регистра WDOGITCR

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 1	–	Зарезервировано
0	Integration Test Module Enable – включение модуля интеграционного тестирования.	Когда установлен в «1», переводит сторожевой таймер в состояние тестирования.

3.14.2.9 WDOGITOP

Регистр Watchdog Integration Test Output Set Register (WDOGITOP) доступен только для записи. В режиме тестирования данные, находящиеся в данном регистре напрямую управляют выходными портами сброса и прерываний.

Описание регистра WDOGITOP приведено в таблице 3.14.7.

Описание регистра WDOGITCR приведено в таблице 3.14.6.				
Таблица 3.14.6 — Описание регистра WDOGITCR				
№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений		
31 – 1	–	Зарезервировано		
0	Integration Test Module Enable – включение модуля интеграционного тестирования.	Когда установлен в «1», переводит сторожевой таймер в состояние тестирования.		
3.14.2.9 WDOGITOP				
Регистр Watchdog Integration Test Output Set Register (WDOGITOP) доступен только для записи. В режиме тестирования данные, находящиеся в данном регистре напрямую управляют выходными портами сброса и прерываний.				
Описание регистра WDOGITOP приведено в таблице 3.14.7.				

Таблица 3.14.7 — Описание регистра WDOGITOR

№ бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 2	Зарезервировано.
1	Выходное значение порта WDOGINT в режиме тестирования.
0	Выходное значение порта WDOGRES в режиме тестирования.

3.15 Передатчик телеметрической информации

3.15.1 Описание структуры передатчика телеметрической информации

Структурная схема передатчика телеметрической информации приведена на рисунке 3.15.1.

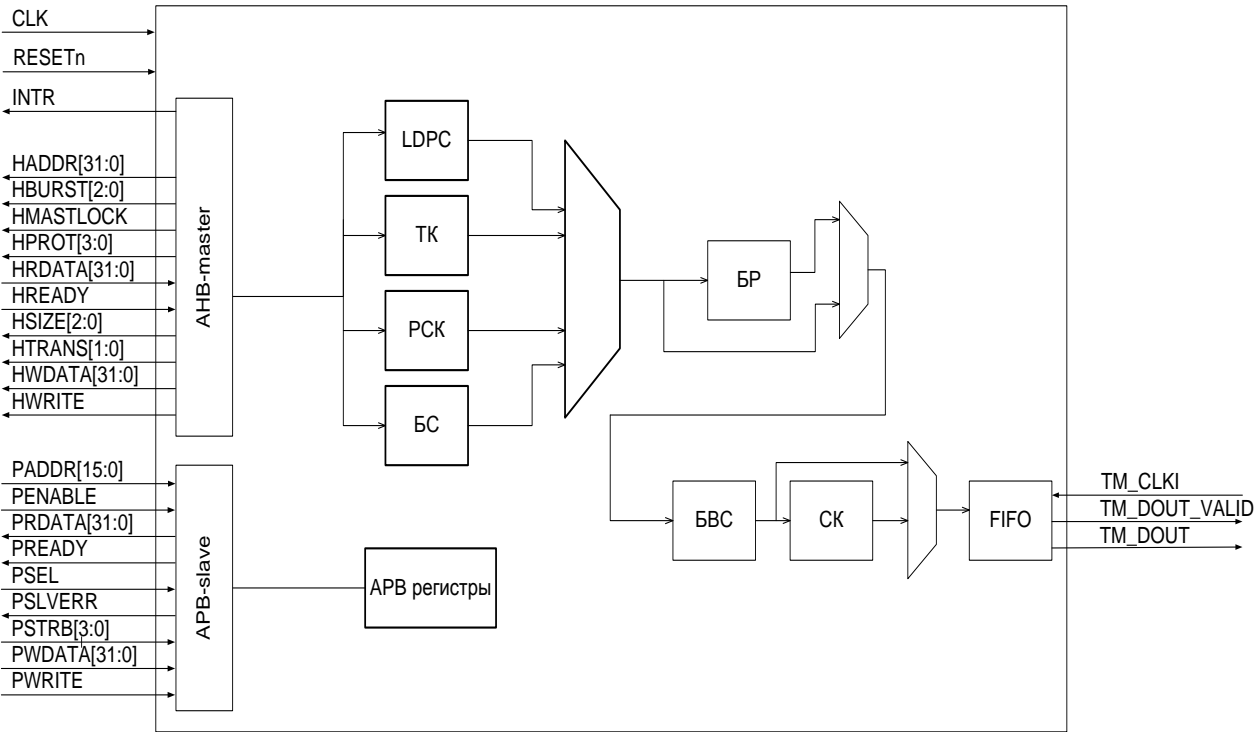


Рисунок 3.15.1 — Структурная схема ПдТМИ

ПдТМИ состоит из следующих составляющих блоков:

- блок кодеров LDPC;
- турбо-кодер (TK);
- кодер Рида-Соломона (PCK);
- блок сериалайзера (BC). Передатчик телеметрии позволяет передавать битовый поток напрямую без кодирования, при этом используется блок сериалайзера;

- блок рандомизатора (БР);
- блок вставки синхромаркера (БВС);
- сверточный кодер (СК).

3.15.2 Описание работы интерфейса передатчика телеметрической информации

Функциональный блок передатчика телеметрической информации позволяет преобразовывать данные из определённой области пользовательской памяти с применением различных методов помехозащищённого кодирования и передавать преобразованные данные в виде битового потока на выходную шину.

Данные из пользовательской памяти передаются побайтно на вход блока младшим битом вперед с помощью мастера на шине AMBA АНВ, который управляется посредством настройки двух дескрипторов. В каждом дескрипторе задается адрес начала области данных для передачи и количество передаваемых данных и затем подается команда на включение дескриптора. В то время, пока один дескриптор осуществляет передачу данных на кодеры передатчика, второй дескриптор доступен для настройки. Таким образом достигается непрерывность передаваемого битового потока. По завершению работы дескриптора вызывается прерывание, после чего возможна повторная настройка дескриптора для последующей передачи.

Кодирование битового потока осуществляется одним из алгоритмов помехозащищенного кодирования (LDPC-кодер, Turbo-кодер, кодер Рида-Соломона). Кодированный поток поступает на блок рандомизатора, далее на блок вставки синхромаркера, далее на сверточный кодер. При этом программная настройка позволяет пропускать данные насквозь без использования кодирования, рандомизации или без свёрточного кодирования.

Структура блоков помехозащищенного кодирования и рандомизатора подробно описана в соответствующих разделах (п. 3.15.2.1 – 3.15.2.5).

В соответствии со стандартом CCSDS данные телеметрии передаются в виде фреймов определённого размера. Перед началом каждого фрейма в битовом потоке

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	<p>дескриптор доступен для настройки. Таким образом достигается непрерывность передаваемого битового потока. По завершению работы дескриптора вызывается прерывание, после чего возможна повторная настройка дескриптора для последующей передачи.</p>	
					<p>Кодирование битового потока осуществляется одним из алгоритмов помехозащищенного кодирования (LDPC-кодер, Turbo-кодер, кодер Рида-Соломона). Кодированный поток поступает на блок рандомизатора, далее на блок вставки синхромаркера, далее на сверточный кодер. При этом программная настройка позволяет пропускать данные насквозь без использования кодирования, рандомизации или без свёрточного кодирования.</p>	
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	<p>Структура блоков помехозащищённого кодирования и рандомизатора подробно описана в соответствующих разделах (п. 3.15.2.1 – 3.15.2.5).</p>	
					<p>В соответствии со стандартом CCSDS данные телеметрии передаются в виде фреймов определённого размера. Перед началом каждого фрейма в битовом потоке</p>	
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	<p>Микросхема интегральная 5023BC016</p>	Лист
					<p>Техническое описание</p>	
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	<p>Версия 2.04 от 01.09.2017 г.</p>	198

необходима вставка синхромаркера в виде заданной битовой последовательности. Блок вставки синхромаркера (БВС) осуществляет добавление этой последовательности с настраиваемым периодом, длиной и содержимым.

После прохождения через блоки кодирования, данные поступают на выходной буфер типа FIFO, откуда передаются на внешнюю шину при наличии входного тактового сигнала TM_CLKI. Отслеживать состояние заполненности буфера можно по специальному флагу, выставление которого свидетельствует о том, что буфер пуст. Размер выходного буфера FIFO равен 256 бит.

Передача данных осуществляется по однобитной шине TM_DOUT и тактируется внешним тактовым сигналом TM_CLKI. По возрастающему фронту сигнала TM_CLKI на шину TM_DOUT выдается следующий бит кодированного потока данных и сигнал TM_DOUT_VALID принимает значение «высокий уровень». Логическая «1» кодируется на шине высоким уровнем, а логический «0» - низким уровнем сигнала. Максимальное значение частоты входного тактового сигнала FTM_CLKI, определяющей скорость выходного битового потока, зависит от включенных кодеков, но при любых настройках не должно быть более половины системной частоты FSYS

$$FTM_CLKI \leq FSYS/2. \quad (10)$$

Информация о выходном интерфейсе физического уровня обобщена в таблице 3.15.1.

Таблица 3.15.1 — Выходной интерфейс передатчика телеметрической информации

Наименование сигнала	Направление	Функциональное назначение
TM_CLKI	Вход	Входной тактовый сигнал
TM_DOUT_VALID	Выход	Признак наличия данных на шине
TM_DOUT	Выход	Выходные данные, обновляются по возрастающему фронту тактового сигнала

Временная диаграмма выходных сигналов изображена на рисунке 3.15.2.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>					Лист
Изм.	Лист	№ докум.	Подп.	Дата						199

3.15.2.3 LDPC-кодер

Low Density Parity Check Code (LDPC-кодер) реализует алгоритм помехоустойчивого кодирования LDPC NASA CCSDS C2 (8160, 7136) и полностью совместим со стандартом GSFC–STD–9100 (Low Density Parity Check Code for Rate 7/8).

LDPC энкодер состоит из блока управления и блока умножения на генерирующую матрицу. Структура блока умножения на генерирующую матрицу показана на рисунке 3.15.4.

В блоке умножения последовательные входные данные умножаются на генерирующую матрицу для получения проверочных данных. Проверочные данные на выходе кодера добавляются к соответствующим входным данным.

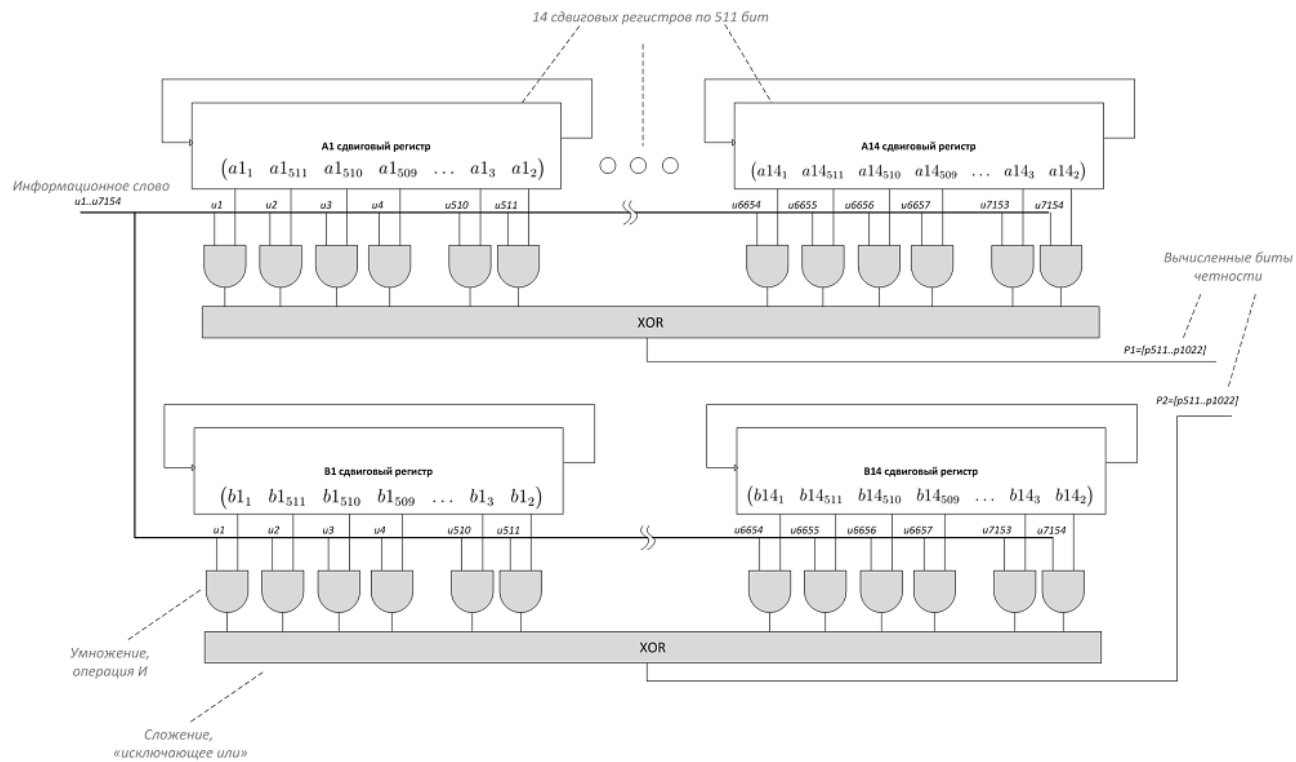


Рисунок 3.15.4 — Структура блока умножения на генерирующую матрицу

3.15.2.4 Рандомизатор

Рандомизатор (блок наложения ПСП) выполняет кодирование данных в соответствии со стандартами CCSDS-131.0-B-2. Блок выполняет скремблирование — обратимое преобразование цифрового потока без изменения скорости передачи с целью получения свойств случайной последовательности.

Име. № подл.	Подп. и дата.
Взам. инв. №	Инв. № дубл.
Подп. и дата	
Име. № подл.	

					Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист
						202
Изм	Лист	№ докум.	Подп.	Дата		

После скремблирования появление «1» и «0» в выходной последовательности равновероятны. Рандомизация является обратимым процессом, исходное сообщение можно восстановить, применив обратный алгоритм.

Перед началом работы блока генерируется псевдослучайная последовательность бит.

Вновь поступающий в рандимизатор бит суммируется по модулю два с битом псевдослучайной последовательности, после чего бит отправляется на выход, скремблер берет следующий входной бит псевдослучайной и входной последовательности и повторяет операцию. Таким образом, в простейшем случае скремблер может быть собран на элементах «исключающее ИЛИ» и триггерах. Обратное преобразование осуществляется в обратном порядке. Псевдослучайная последовательность используется циклически.

В соответствии со стандартом, полиномом рандомизатора является: $h(x) = x^8 + x^7 + x^5 + x^3 + 1$, используемый как в виде линейного сдвигового регистра с обратными связями Фибоначчи, инициализируемой между фреймами.

Структурная схема рандомизатора показана на рисунке 3.15.5.

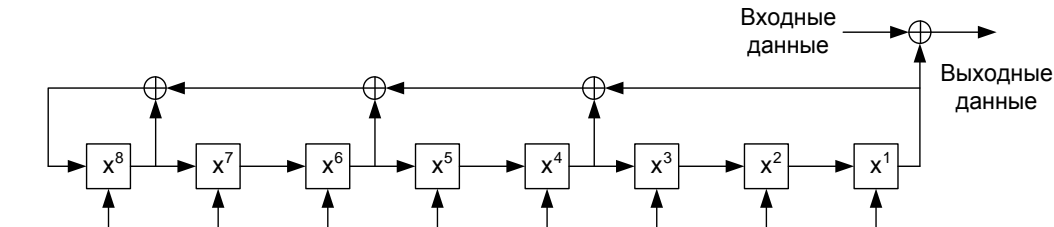


Рисунок 3.15.5 — Структурная схема рандомизатора

3.15.2.5 Сверточный кодер

Сверточный кодер в контроллере передатчика телеметрической информации реализуется двумя схемами: с прореживанием и без прореживания.

Структурная схема сверточного кодера без прореживания изображена на рисунке 3.15.6.

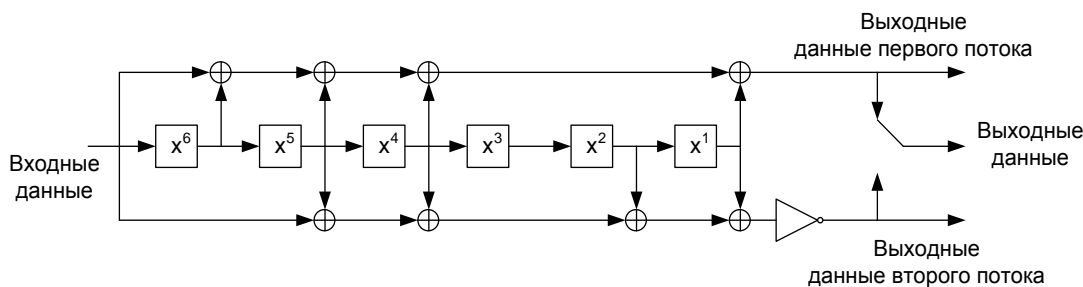


Рисунок 3.15.6 — Структура сверточного энкодера без прореживания

Схема сверточного кодера с прореживанием описывается в стандарте CCSDS-131.0-B-1. Четыре различных схемы прореживания битов предоставляют выбор большинства необходимых скоростей кодирования ($2/3$, $3/4$, $5/6$ или $7/8$) и соответствующих уровней коррекции ошибок.

Структурная схема сверточного кодера с прореживанием изображена на рисунке 3.15.7.

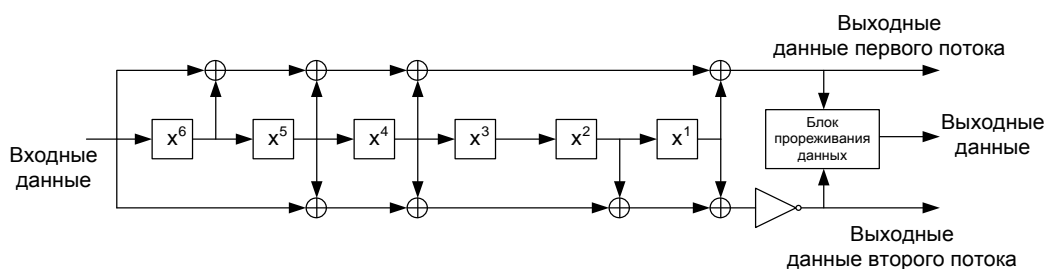


Рисунок 3.15.7 — Структура сверточного кодера с прореживанием

Образующими многочленами обоих сверточных кодеров являются $G1 = 1111001$ (171 ОСТ) и $G2 = 1011011$ (133 ОСТ).

Блоком поддерживается перфорация до скоростей $2/3$, $3/4$, $5/6$ и $7/8$. Порядок следования битов:

- при скорости $2/3$: $C1(1)$, $C2(1)$, $C2(2)$...
- при скорости $3/4$: $C1(1)$, $C2(1)$, $C2(2)$, $C1(3)$...
- при скорости $5/6$: $C1(1)$, $C2(1)$, $C2(2)$, $C1(3)$, $C2(4)$, $C1(5)$...
- при скорости $7/8$: $C1(1)$, $C2(1)$, $C2(2)$, $C2(3)$, $C2(4)$, $C1(5)$, $C2(6)$, $C1(7)$...

где $C1(n)$ и $C2(n)$ — биты, сформированные при помощи многочленов $G1$ и $G2$ соответственно.

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	Выходные данные второго потока		
<p>Рисунок 3.15.7 — Структура сверточного кодера с прореживанием</p> <p>Образующими многочленами обоих сверточных кодеров являются $G1 = 1111001$ (171 ОСТ) и $G2 = 1011011$ (133 ОСТ).</p> <p>Блоком поддерживается перфорация до скоростей $2/3$, $3/4$, $5/6$ и $7/8$. Порядок следования битов:</p> <ul style="list-style-type: none">— при скорости $2/3$: $C1(1)$, $C2(1)$, $C2(2)$...— при скорости $3/4$: $C1(1)$, $C2(1)$, $C2(2)$, $C1(3)$...— при скорости $5/6$: $C1(1)$, $C2(1)$, $C2(2)$, $C1(3)$, $C2(4)$, $C1(5)$...— при скорости $7/8$: $C1(1)$, $C2(1)$, $C2(2)$, $C2(3)$, $C2(4)$, $C1(5)$, $C2(6)$, $C1(7)$... <p>где $C1(n)$ и $C2(n)$ — биты, сформированные при помощи многочленов $G1$ и $G2$ соответственно.</p>							
					Микросхема интегральная 5023BC016		Лист
					Техническое описание		
					Версия 2.04 от 01.09.2017 г.		204
Изм.	Лист	№ докум.	Подп.	Дата			

3.15.3 Программная модель управления передатчиком телеметрической информации

Для передачи телеметрической информации необходимо осуществить предварительную настройку блоков кодирования, затем настроить и включить дескрипторы в работу. Управление блоком осуществляется через запись и чтение регистров на шине APB. Последовательность действий в общем случае описана ниже:

— отключить блоки кодирования записью в регистр GLOBAL_ENABLE значения «0». Это необходимо для того, чтобы вносимые изменения вступили в силу после включения блоков, т.к. обновление параметров блоков происходит по переходу GLOBAL_ENABLE из «0» в «1»;

— с помощью регистра CODER_SEL включить в работу необходимый блок помехозащищённого кодирования, рандомизатора, сверточный кодер;

— произвести настройку кодирующих блоков в регистре CODER_CONFIG. Каждый блок может работать в одном из нескольких режимов, поддерживаемых стандартом;

— произвести настройку синхромаркера. В регистре MARKER_AND_FRAME_SIZE необходимо настроить период вставки синхромаркера и его размер в битах (не более 192 бит). Для работы в соответствии со стандартом CCSDS необходимо настроить период вставки синхромаркера (размер фрейма) в зависимости от выбранных кодеков. То есть, задаваемый период должен быть равен размеру фрейма после проведения кодирования (таблица 3.15.2). Регистры MARKER1 – MARKER6 определяют содержимое синхромаркера;

Таблица 3.15.2 — Данные для определения размеров фрейма в зависимости от выбранного кодера и его режима работы

Тип кодера	Размер фрейма информационных данных на входе энкодера (в байтах)	Размер фрейма на выходе энкодера (в битах)
1	2	3
Reed-Solomon (255,239)	239	255x8 = 2040
Reed-Solomon (255,223)	223	255x8 = 2040

Име. № подл.	Подп. и дата.	Име. № дубл.	Подп. и дата.	Взам. инв. №	Име. №	Изм	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 205
--------------	---------------	--------------	---------------	--------------	--------	-----	------	----------	-------	------	---	-------------

Продолжение таблицы 3.15.2

1	2	3
Turbo coder, rate 1/2	223	$223 \times 8 \times 2 + 4 \times 2 = 3576$
Turbo coder, rate 1/3	223	$223 \times 8 \times 3 + 4 \times 3 = 5364$
Turbo coder, rate 1/4	223	$223 \times 8 \times 4 + 4 \times 4 = 7152$
Turbo coder, rate 1/6	223	$223 \times 8 \times 6 + 4 \times 6 = 10728$
LDPC coder (8160 ,7136)	892	8160

— настроить дескрипторы на передачу. Для управления каждым дескриптором используется два регистра DSCR_x_PTR и DSCR_x_FLAG_SIZE (x = 1, 2). Регистр DSCR_x_PTR задает адрес начала области данных в пользовательской памяти, с которого начнется вычитывание данных для передачи на блоки кодирования. Регистр DSCR_x_FLAG_SIZE определяет количество вычитываемых байт (от 1 до 2048) и включает дескриптор в работу;

— запустить блоки кодирования в работу записью в GLOBAL_ENABLE значения «1». После этого открытые дескрипторы начнут вычитывание данных. Первым начинает работу дескриптор с номером 1. Также после включения следует сбросить флаг опустошения FIFO путём записи в регистр PHY_FIFO_UNDERRUN значения «1». Это позволит отслеживать состояние передающего буфера во время работы;

— по завершению работы дескриптора вызывается прерывание. Для обоих дескрипторов используется один общий вектор прерывания. Определить, какой из дескрипторов завершил работу и готов для дальнейшей настройки, можно по состоянию бита DATA_PRESENCE в регистре DSCR_x_FLAG_SIZE. Если необходимо передавать поток данных большего размера, чем может быть помещён в два дескриптора, потребуется повторное открытие дескриптора после его закрытия. При этом для обеспечения непрерывности битового потока на выходе блока дескриптор должен быть повторно настроен до того, как второй дескриптор завершит работу;

— после того, как все необходимые данные были переданы из пользовательской памяти и отправлены на физический уровень, будет выставлен в «1» флаг PHY_FIFO_UNDERRUN. Это свидетельствует о том, что блоки кодирования полностью завершили обработку и передачу данных. Если флаг

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>					Лист
Изм.	Лист	№ докум.	Подп.	Дата						206

выставился до того, как пользователь передал все необходимые данные, это значит, что нарушилась непрерывность передачи данных вследствие того, что дескрипторы передавали данные в FIFO с меньшей скоростью, чем данные выдавались на выходную шину.

3.15.4 Регистры передатчика телеметрической информации

Перечень регистров передатчика телеметрической информации приведён в таблице 3.15.3. Описание регистров общего назначения приведено в таблицах 3.15.4 — 3.15.11.

Таблица 3.15.3 — Перечень регистров передатчика телеметрической информации

Смещение	Название регистра	Тип	Значение по сбросу	Описание
0x00	DSCR1_PTR	RW	0	Дескриптор 1: указатель на область памяти.
0x04	DSCR1_FLAG_SIZE	RW	0	Дескриптор 1: флаги и размер области памяти.
0x08	DSCR2_PTR	RW	0	Дескриптор 2: указатель на область памяти.
0x0C	DSCR2_FLAG_SIZE	RW	0	Дескриптор 2: флаги и размер области памяти.
0x10	GLOBAL_ENABLE	RW	0	Включение блока в режим передачи.
0x14	CODER_SEL	RW	0	Выбор энкодеров.
0x18	PHY_FIFO_UNDERRUN	RW	0	Флаг опустошения FIFO.
0x1C	MARKER_AND_FRAME_SIZE	RW	0	Размер фрейма и размер синхромаркера.
0x20	MARKER1	RW	0	Содержимое синхромаркера.
0x24	MARKER2	RW	0	Содержимое синхромаркера.
0x28	MARKER3	RW	0	Содержимое синхромаркера.
0x2C	MARKER4	RW	0	Содержимое синхромаркера.
0x30	MARKER5	RW	0	Содержимое синхромаркера.
0x34	MARKER6	RW	0	Содержимое синхромаркера.
0x38	CODERS_CONFIG	RW	0	Настройки энкодеров.
0x3C-0xFFFF	—			Не используется, читается 0.

Ине. № подл.	Подп. и дата	Взам. инв. №	Ине. № дубл.	Подп. и дата.

3.15.4.1 Регистры DSCR1_PTR, DSCR2_PTR

Таблица 3.15.4 — Описание регистров DSCR1_PTR, DSCR2_PTR

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 2	DSCR_PTR	указатель на начало передаваемой области данных во внутренней памяти. Начало области должно быть выровнено по 4-байтной границе, младшие 2 бита указателя нули
1 – 0		

3.15.4.2 Регистры DSCR1_FLAG_SIZE, DSCR2_FLAG_SIZE

Таблица 3.15.5 — Описание регистров DSCR1_FLAG_SIZE, DSCR2_FLAG_SIZE

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31	DATA_PREP (RW)	следует записать «1», когда данные этого дескриптора готовы для передачи. Когда GLOBAL_ENABLE = 1, то данный дескриптор будет задействован, и данные, на которые ссылается этот дескриптор, будут считываться из внутренней памяти и записываться во внутренние буферы энкодеров. В этот момент изменение полей DSCR_PTR и DATA_SIZE дескриптора не оказывает влияния. Когда все данные считаются и запишутся во внутренние буферы, в поле DATA_PREP будет аппаратно записан 0.
30 – 11	—	
10 – 0	DATA_SIZE (RW)	размер передаваемой области данных в байтах минус 1. Например, если 0, то передаём 1 байт, если 1, то передаём 2 байта, если 2047 — передаём 2048 байт.

Примечание — Если установлены флаги DATA_PREP в двух дескрипторах, и аппаратный блок не работает ни с одним из них (это может быть только в первый момент после включения блока через GLOBAL_ENABLE), то первым для передачи выбирается дескриптор номер 1 (DSCR1).

3.15.4.3 Регистр GLOBAL_ENABLE

Таблица 3.15.6 — Описание регистра GLOBAL_ENABLE

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 1	—	
0	GLOBAL_ENABLE (RW)	0 — все энкодеры, FIFO, AHB контроллер находятся в асинхронном сбросе. 1 — включение всех модулей, ответственных за передачу данных.

Примечание — Конфигурация энкодеров, которая задается через регистры CODER_SEL, MARKER_AND_FRAME_SIZE, MARKER1-6, CODERS_CONFIG,

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.					Лист
					Изм	Лист	№ докум.	Подп.	Дата	
										208

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.

Таблица 3.15.7 — Описание регистра PHY_FIFO_UNDERRUN

Примечание — В самом начале работы при переходе GLOBAL_ENABLE из нуля в единицу, флаг устанавливается в «1», необходимо подождать, пока данные будут подаваться в выходной блок FIFO и далее сбросить флаг.

Таблица 3.15.8 — Описание регистра CODER_SEL

					Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист
Изм.	Лист	№ докум.	Подп.	Дата		209

3.15.4.6 Регистр MARKER_AND_FRAME_SIZE

Таблица 3.15.9 — Описание регистра MARKER_AND_FRAME_SIZE

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 24	–	
23 – 16	MARKER_SIZE (RW)	размер тела синхромаркера в битах
15 – 0	FRAME_SIZE (RW)	период вставки синхромаркера в битах

Примечания:

1. Размер фрейма определяется типом используемого помехозащищённого кодера. Пользователь должен высчитать размер фрейма с учётом дополнительных бит создаваемых энкодером, и записать нужное значение в поле FRAME_SIZE. Таблица 3.15.9 содержит данные для определения размеров фрейма в случае применения всевозможных типов кодеров.

2. Если размер маркера равен 0, то синхромаркер не вставляется.

3.15.4.7 Регистры MARKER1 – MARKER6

Таблица 3.15.10 — Описание регистров MARKER1 - MARKER6

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 0	MARKER (RW)	содержимое синхромаркера. Первым передается бит с номером 0 регистра MARKER1, последним, если размер синхромаркера выставлен 192 бита, 31-ый бит регистра MARKER6.

3.15.4.8 Регистр CODERS_CONFIG(RW)

Таблица 3.15.11 — Описание регистра EXTMEM_CTRL

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
1	2	3
31 – 25	–	
24	LDPC_Type	Тип LDPC энкодера: 0 — (8160 ,7136); Для корректной работы в этом поле должно быть значение 0.
23 – 18	–	
17 – 16	RS_Type	тип энкодера RS: [16] 0 — (255,239); 1 — (255,223); [17] 0 — обычный; 1 — dual basis converted.
15 – 10	–	

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.					
Изм	Лист	№ докум.	Подп.	Дата					
					Микросхема интегральная 5023BC016				
					Техническое описание				
					Версия 2.04 от 01.09.2017 г.				
					Лист				
					210				

Продолжение таблицы 3.15.11

1	2	3
9 – 8	Turbo_rate	тип Turbo кодера: 0 — rate 1/2; 2 — rate 1/4; 1 — rate 1/3; 3 — rate 1/6.
7 – 3	—	
2 – 0	Conv_rate	тип сверточного энкодера: 2 — rate 2/3; 3 — rate 3/4; 5 — rate 5/6; 7 — rate 7/8; остальные значения — rate 1/2.

3.16 Приёмник телекомандной информации

3.16.1 Общее описание приёмника телекомандной информации

Структурная схема приёмника телекомандной информации приведена на рисунке 3.16.1.

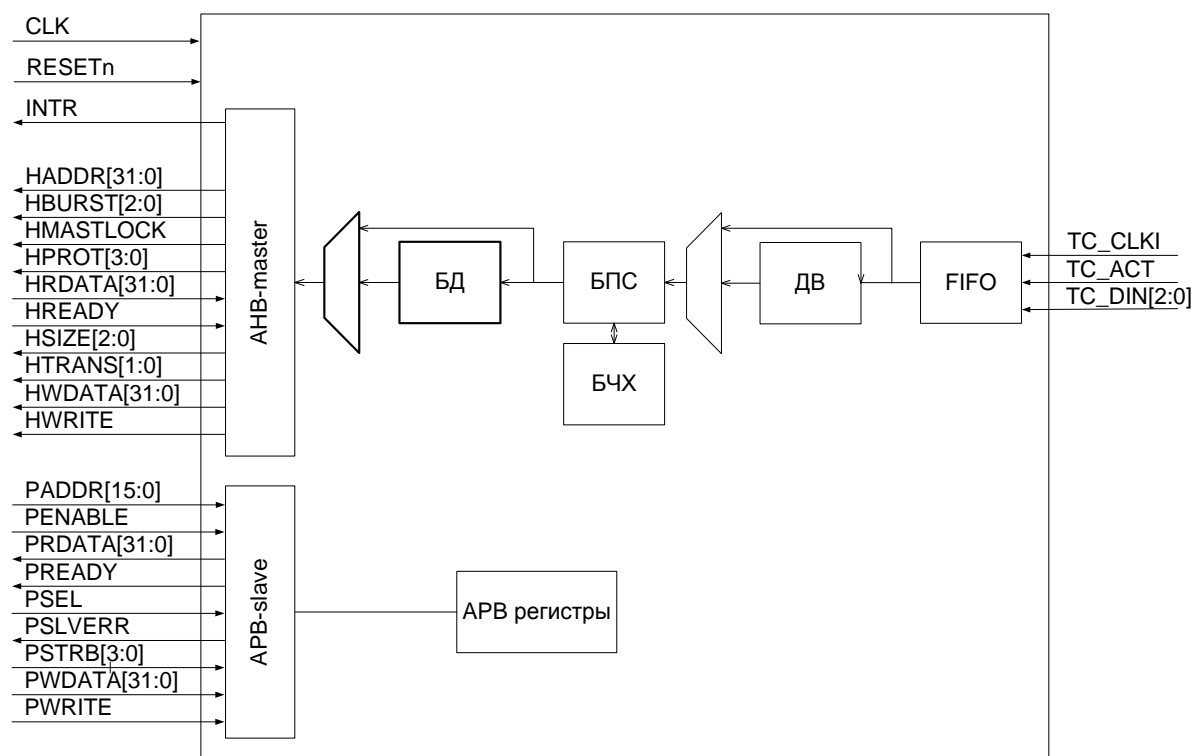


Рисунок 3.16.1 — Структурная схема ПрТКИ

Приёмник телекомандной информации состоит из следующих функциональных блоков:

- декодера Витерби;
- блока поиска синхромаркера (БПС);

Име. № подл.	Взам. инв. №	Инв. № дубл.	Подп. и дата.
Изм.	Лист	№ докум.	Подп.
Дата			

Микросхема интегральная 5023BC016

Техническое описание
Версия 2.04 от 01.09.2017 г.

Лист

211

Копировал

Формат А4

- декодера Боуза-Чоудхури-Хоквингема (БЧХ);
- блока дерандомизатора (БД).

3.16.2 Описание работы интерфейса приёмника телекомандной информации

Функциональный блок приёмника телекомандной информации позволяет принимать битовый поток данных, закодированных в соответствии со стандартом CCSDS, осуществлять его декодирование и сохранять полученные данные в пользовательской памяти.

Интерфейс физического уровня содержит сигнал входного тактового сигнала TC_CLKI, сигнал активности данных TC_ACT и шину данных TC_DIN. Логическая «1» на шине данных передаётся в виде высокого уровня сигнала, логический «0» — в виде низкого уровня. Данные с шины считываются в блок по возрастающему фронту сигнала TC_CLKI. Высокий уровень сигнала TC_ACT свидетельствует о наличии данных на шине в момент прихода возрастающего фронта сигнала TC_CLKI. Для корректной работы декодеров блока частота тактового входного сигнала FTC_CLKI должна быть меньше системной тактовой частоты FSYS в соответствии с условием

$$FTC_CLKI \leq 0,4 \cdot FSYS. \quad (11)$$

Информация о входном интерфейсе физического уровня обобщена в таблице 3.16.1.

Таблица 3.16.1 — Входной интерфейс приёмника телекомандной информации

Наименование сигнала	Направление	Функциональное назначение
TC_CLKI	Вход	Входной тактовый сигнал
TC_ACT	Вход	Признак наличия данных на шине
TC_DIN	Вход	Входные данные, считываются по возрастающему фронту тактового сигнала

Временная диаграмма сигналов входного интерфейса изображена на рисунке 3.16.2.

Изм.	Лист	№ докум.	Подп.	Дата	частоты FSYS в соответствии с условием		
					$FTC_CLKI \leq 0,4 \cdot FSYS.$		
					(11)		
					Информация о входном интерфейсе физического уровня обобщена в		
					таблице 3.16.1.		
Таблица 3.16.1 — Входной интерфейс приёмника телекомандной информации							
Наименование сигнала		Направление		Функциональное назначение			
TC_CLKI		Вход		Входной тактовый сигнал			
TC_ACT		Вход		Признак наличия данных на шине			
TC_DIN		Вход		Входные данные, считываются по возрастающему фронту тактового сигнала			
Временная диаграмма сигналов входного интерфейса изображена на рисунке 3.16.2.							
					Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.		Лист 212

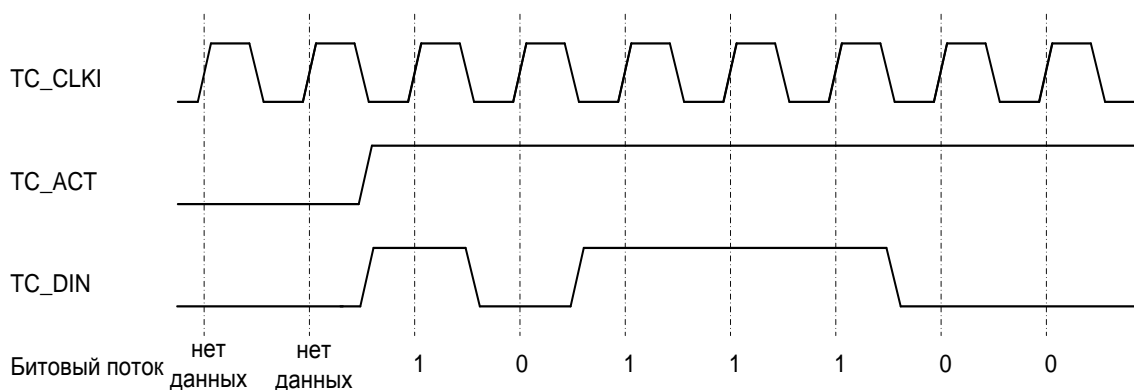


Рисунок 3.16.2 — Входной интерфейс приёмника телекомандной информации

Буфер FIFO, получающий входной битовый поток, имеет размер 256 бит.

Входной битовый поток поступает из входного FIFO на блок поиска синхромаркера либо напрямую, либо после прохождения через сверточный декодер Витерби. В соответствии со стандартом синхромаркер представляет из себя фиксированную последовательность из 16 бит, значение которой устанавливается программно. Если во входном потоке встречается указанная последовательность, это трактуется блоком БПС как начало фрейма, и он передаётся для дальнейшей обработки.

Декодер БЧХ позволяет найти и исправить одну битовую ошибку на 7 байт данных из поступающего битового потока. Каждый восьмой байт в принимаемом фрейме используется в качестве контрольной суммы для поиска и исправления битовой ошибки в предыдущих 7 байтах. В связи с такой логикой работы размер полезных данных в каждом фрейме должен быть кратен 7 байтам.

Если БЧХ декодер обнаруживает неисправимую ошибку (что должно проявляться в случае, если в пакете из 7 байт искажено более 2 бит данных), на этом приём фрейма завершается, при этом в принимающем дескрипторе выставляется флаг ошибки (см. описание ниже).

Исправленные данные поступают на блок дерандомизатора, который производит дескремблирование данных.

В конце каждого принимаемого фрейма должна присутствовать хвостовая последовательность бит, имеющая размер 64 бита. Приём такой последовательности

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	<p>Декодер БЧХ позволяет найти и исправить одну битовую ошибку на 7 байт данных из поступающего битового потока. Каждый восьмой байт в принимаемом фрейме используется в качестве контрольной суммы для поиска и исправления битовой ошибки в предыдущих 7 байтах. В связи с такой логикой работы размер полезных данных в каждом фрейме должен быть кратен 7 байтам.</p> <p>Если БЧХ декодер обнаруживает неисправимую ошибку (что должно проявляться в случае, если в пакете из 7 байт искажено более 2 бит данных), на этом приём фрейма завершается, при этом в принимающем дескрипторе выставляется флаг ошибки (см. описание ниже).</p> <p>Исправленные данные поступают на блок дерандомизатора, который производит дескремблирование данных.</p> <p>В конце каждого принимаемого фрейма должна присутствовать хвостовая последовательность бит, имеющая размер 64 бита. Приём такой последовательности</p>	
Изм.	Лист	№ докум.	Подп.	Дата	<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>	Лист
						213

свидетельствует об окончании приёма фрейма. Блок приёма телекомандной информации никак не ограничивает максимальный размер принимаемого фрейма.

Передача декодированных данных в область пользовательской памяти осуществляется за счёт системы дескрипторов. Поддерживается до шестнадцати дескрипторов, каждый из которых при открытии позволяет принять от 1 до 1024 байт декодированных данных. При приёме битового потока дескрипторы используются последовательно, то есть при завершении работы дескриптора с номером N, активен становится дескриптор с номером N + 1, если он был до этого настроен на работу. Если же этот дескриптор не был открыт к моменту переключения на него, приём данных остановится и при наличии данных на входном интерфейсе произойдет переполнение входного FIFO.

После закрытия дескриптора с максимальным возможным номером (равным 16) активен становится дескриптор с минимальным номером. Если пользователь решает использовать не весь набор дескрипторов, то переключение между ними можно производить программно. Номер активного дескриптора можно изменить, но только в момент, когда текущий активный дескриптор не осуществляет приём данных. По умолчанию активным является дескриптор с наименьшим номером.

Если размер принимаемого фрейма превышает размер активного дескриптора, по заполнению дескриптор будет закрыт, а данные продолжат передаваться с помощью следующего дескриптора. В этом случае для анализа принятых данных в дескрипторах используются флаги, с помощью которых можно определить, что дескриптор принял начало или конец пакета.

По закрытию каждого дескриптора происходит вызов прерывания. Блоком приёма телекоманд используется единственный вектор прерывания.

3.16.2.1 Декодер Витерби

Декодер Витерби предназначен для восстановления информационного битового потока цифровых данных, закодированных сверточным кодом.

Ине. № подл.	Подп. и дата.
Взам. инв. №	Инв. № дубл.
Подп. и дата	
Ине. № подл.	

					Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 214
Изм	Лист	№ докум.	Подп.	Дата		

Декодер Витерби состоит из следующих функциональных блоков:

- а) перемежитель;
- б) блок вычисления метрик;
- в) блок памяти вероятных последовательностей;
- г) селектор выходных данных;
- д) блок управления.

На рисунке 3.16.3 представлена структурная схема декодера Витерби.

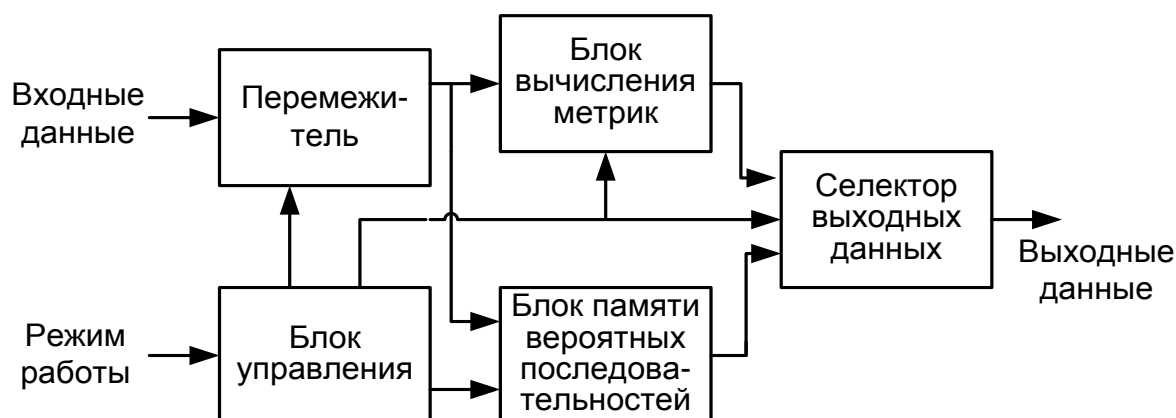


Рисунок 3.16.3 — Структурная схема декодера Витерби

Декодер Витерби обладает следующими характеристиками:

- порождающие полиномы кода — $g_1 = 171\text{ОСТ}$, $g_2 = 133\text{ОСТ}$, $K = 7$;
- скорости декодирования — $1/2$, $2/3$, $3/4$, $5/6$, $7/8$;
- возможность декодирования жестких и трёх- и четырёхуровневых «мягких» решений;
- глубина декодирования — 128 бит.

3.16.2.2 БЧХ декодер

Блок предназначен для декодирования кода БЧХ с порождающим полиномом $g(x) = x^7 + x^6 + x^2 + 1$ согласно стандарту CCSDS 231.0-B-2. Он выполняет вычисление синдрома и исправление одной ошибки путём перебора ее возможных позиций. Блок выдает предупреждение о неисправимой ошибке, если вычисленному синдрому не соответствует ни одна из возможных позиций ошибочного бита.

На рисунке 3.16.4 показана функциональная схема декодера БЧХ.

Инв. № подл.	Подп. и дата.			
	Инв. № дубл.			
	Взам. инв. №			
	Подп. и дата			
Изм.	Лист	№ докум.	Подп.	Дата
Микросхема интегральная 5023BC016				
Техническое описание				
Версия 2.04 от 01.09.2017 г.				
Копировал				Лист
Формат А4				215

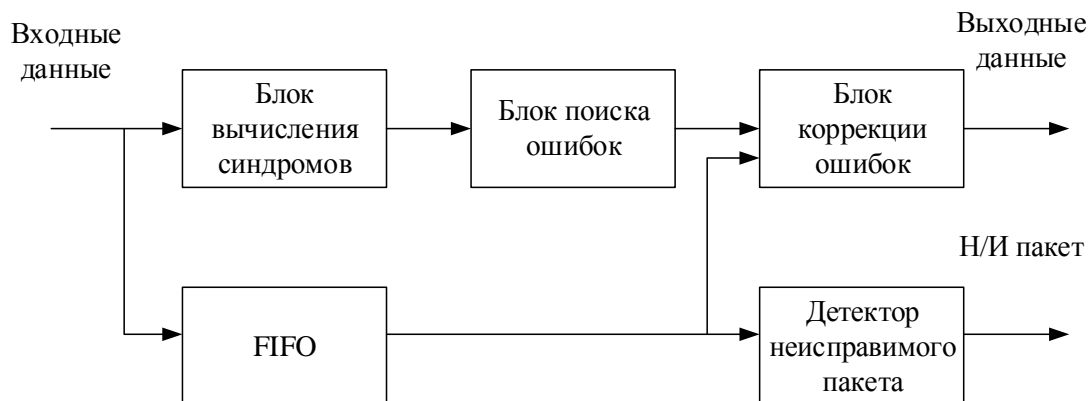


Рисунок 3.16.4 — Функциональная схема декодера БЧХ

3.16.2.3 Дерандомизатор

Дерандомизатор выполняет функцию дескремблирования принимаемого цифрового потока данных.

Механизм дескремблирования основывается на повторном скремблировании цифрового потока при использовании идентичного генератора ПСП и сигналов инициализации скремблера, как на передающей стороне.

Блок выполнен согласно стандарту CCSDS 231.0-B-2 и использует псевдослучайную последовательность с порождающим полиномом $h(x) = x^8 + x^6 + x^4 + x^3 + x^2 + x + 1$. Структурная схема дерандомизатора показана на рисунке 3.16.5.

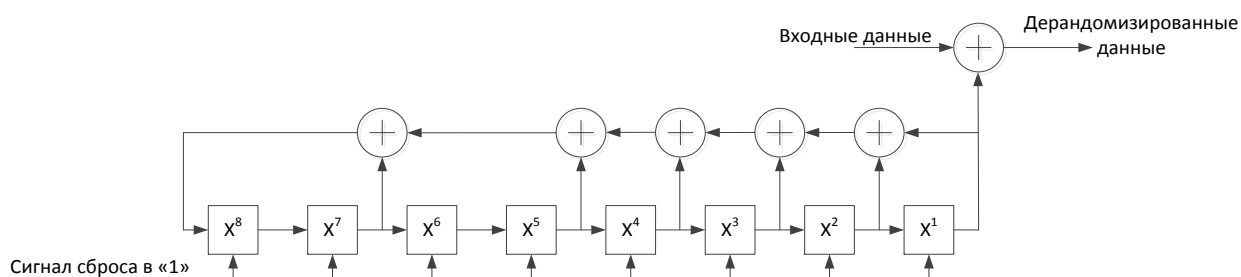


Рисунок 3.16.5 — Структурная схема дерандомизатора

3.16.3 Программная модель управления приёмником телекомандной информации

Для приёма телекомандной информации необходимо осуществить предварительную настройку блоков декодирования, затем настроить и включить дескрипторы в работу. Управление блоком осуществляется через запись и чтение

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.
Изм.	Лист	№ докум.	Подп.	Дата
Микросхема интегральная 5023BC016				
Техническое описание				
Версия 2.04 от 01.09.2017 г.				
Копировал				Лист
Формат А4				216

регистров на шине APB. Последовательность действий в общем случае описана ниже:

а) отключить блоки приёмника телекоманд записью в регистр GLOBAL_ENABLE значения «0». Это необходимо для того, чтобы вносимые изменения вступили в силу после включения блоков, т.к. обновление параметров блоков происходит по переходу GLOBAL_ENABLE из «0» в «1»;

б) произвести настройку блока поиска синхромаркера. Содержимое стартовой последовательности задаётся в регистре START_SEQ, содержимое хвостовой последовательности задаётся в регистрах TAIL1_SEQ и TAIL2_SEQ. Путём задания специального бита в регистре START_SEQ можно включить разрешение единичной битовой ошибки в стартовой последовательности. В этом случае последовательность бит, отличающаяся одним разрядом от установленной в регистре, также будет считаться корректной стартовой последовательностью;

в) произвести включение/отключение отдельных функциональных блоков: БЧХ-декодера, рандомизатора. Блоки включаются в соответствующих полях регистра CODER_SEL. Регистр GLOBAL_ENABLE используется для включения декодеров в работу, при этом обновлённые значения из регистров CODER_SEL окажут влияние только по переходу содержимого GLOBAL_ENABLE из «0» в «1»;

г) запустить блоки приёмника телекоманд в работу записью в GLOBAL_ENABLE значения «1»;

д) задать номер активного дескриптора в регистре DSCR_CURRENT. Приём данных начнётся с данного дескриптора;

е) включить необходимые дескрипторы в работу. Для включения каждого дескриптора используется по два регистра. Регистр DSCR_PTRx используется для указания начала области пользовательской памяти, в которую будет производиться запись декодированных данных. Для открытия дескриптора регистр DSCR_FLAG_SIZEx используется следующим образом:

- 1) задается размер дескриптора в поле DATA_SIZE (от 1 до 1024 Б);
- 2) дескриптор включается в работу записью «1» в поле DSCR_READY;

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.						
Изм	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016					Лист
					Техническое описание					
					Версия 2.04 от 01.09.2017 г.					217

3) для работы с большими объёмами входных данных, поступающих непрерывно, рекомендуется использовать все 16 дескрипторов, каждый из которых предварительно открывается на приём максимального количества данных (1024 байта);

ж) по мере поступления данных на вход блока приёмника телекоманд активный в настоящий момент дескриптор будет производить запись данных в пользовательскую память после прохождения их через декодирующие модули. Дескриптор будет закрыт в одном из случаев:

1) количество принятых данных превысило выделенный в дескрипторе размер;

2) фрейм завершился до того, как количество принятых данных превысило выделенный в дескрипторе размер;

3) декодером БЧХ была обнаружена неисправимая ошибка при приёме пакета;

4) в каждом из перечисленных случаев управление будет передано следующему по номеру дескриптору;

5) если следующий дескриптор в этот момент открыт, поступающие далее на вход данные продолжат записываться в пользовательскую память в соответствии с указанным в дескрипторе адресом;

6) если же следующий дескриптор закрыт, данные не будут переданы в пользовательскую память, и, если в буфере FIFO закончится место для приёма последующих данных, будет выставлен флаг переполнения FIFO в регистре PHY_FIFO_OVERFLOW. После того, как произошло переполнение, дальнейший приём будет корректен только после перезапуска модулей приёмника телекоманд записью в GLOBAL_ENABLE значений «0» и «1» последовательно;

з) после закрытия дескриптора происходят следующие события:

1) выставляется прерывание;

2) в поле DATA_SIZE регистра DSCR_FLAG_SIZE_x записывается количество принятых дескриптором данных;

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	следующему по номеру дескриптору;					
					5) если следующий дескриптор в этот момент открыт, поступающие далее на вход данные продолжают записываться в пользовательскую память в соответствии с указанным в дескрипторе адресом;					
					б) если же следующий дескриптор закрыт, данные не будут переданы в пользовательскую память, и, если в буфере FIFO закончится место для приёма последующих данных, будет выставлен флаг переполнения FIFO в регистре PHY_FIFO_OVERFLOW. После того, как произошло переполнение, дальнейший приём будет корректен только после перезапуска модулей приёмника телекоманд записью в GLOBAL_ENABLE значений «0» и «1» последовательно;					
					з) после закрытия дескриптора происходят следующие события:					
					1) выставляется прерывание;					
Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	2) в поле DATA_SIZE регистра DSCR_FLAG_SIZEx записывается количество принятых дескриптором данных;					
Изм.	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016					Лист
					Техническое описание					
					Версия 2.04 от 01.09.2017 г.					218

DSCR_TBL — таблица дескрипторов — 16 дескрипторов. Каждый дескриптор состоит из двух регистров. Первый регистр DSCR_PTR, второй — DSCR_FLAG_SIZE.

3.16.4.1 Регистры DSCR_PTR

Таблица 3.16.3 — Описание регистра DSCR_PTR

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 2	DSCR_PTR (RW)	указатель на начало области во внутренней памяти, предназначенной для записи принятых данных. Начало области должно быть выровнено по 4 байтной границе, младшие 2 бита указателя — нулевые
1 – 0	–	

3.16.4.2 Регистры DSCR_FLAG_SIZE

Таблица 3.16.4 — Описание регистра DSCR_FLAG_SIZE

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31	DSCR_READY (RW)	нужно программно установить в «1», когда область памяти, на которую указывает этот дескриптор, освобождена и готова для записи принятых данных. Когда все принятые данные записаны в область, на которую указывает этот дескриптор, то в этот бит аппаратно сбрасывается в «0»
30 – 19	–	
18 – 16	DSCR_FLAGS (RW)	[16] — SOF: устанавливается в «1», если записанные принятые данные содержат первый байт фрейма. Используется для случаев, когда принятый фрейм записывается в несколько дескрипторов [17] — EOF: устанавливается в «1», если записанные принятые данные содержат последний байт фрейма. Используется для случаев, когда принятый фрейм записывается в несколько дескрипторов [18] — Error: флаг ошибки устанавливается в «1», если принятый фрейм завершается не маркером хвоста, а неисправимой ошибкой БЧХ-декодера
15 – 10	–	
9 – 0	DATA_SIZE (RW)	Размер области, в которую будут записываться принятые данные. «0» означает, что размер области памяти 1024 Б. Если размер пакета принятых данных меньше размера области, то по окончании записи данных в это поле будет аппаратно записан размер записанных данных. Если размер пакета больше, то поле останется без изменения, а остаток данных запишется в следующий дескриптор

Флаги записываются аппаратно, когда все принятые данные записаны в область, на которую указывает этот дескриптор.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	Микросхема интегральная 5023BC016					Лист
					Техническое описание					
					Версия 2.04 от 01.09.2017 г.					220
					Изм	Лист	№ докум.	Подп.	Дата	

3.16.4.3 Регистр DSCR_CURRENT

Таблица 3.16.5 — Описание регистра DSCR_CURRENT

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31	WORK_FLAG (RO)	устанавливается в «1», если текущий дескриптор находится в работе и устанавливается в «0», когда текущий дескриптор не готов, то есть когда флаг текущего дескриптора DSCR_READY находится в «0».
30 – 4	–	
3 – 0	DSCR_NUM (RW)	номер текущего дескриптора, RO, когда WORK_FLAG равен «1», и автоматически инкрементируется после отработки дескриптора. Когда WORK_FLAG равен «0», то номер дескриптора можно перезаписать программно

3.16.4.4 Регистр GLOBAL_ENABLE

Таблица 3.16.6 — Описание регистра GLOBAL_ENABLE

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 1	–	
0	GLOBAL_ENABLE (RW)	0 — все декодеры, FIFO, АНВ-контроллер находятся в асинхронном сбросе. 1 — включение всех модулей, ответственных за приём данных.

Примечание — Конфигурация декодеров, которая задаётся через регистры CODER_SEL, START_SEQ, TAIL_SEQ1, TAIL_SEQ2 загружается только в момент перехода GLOBAL_ENABLE из нуля в единицу. Если менять значения этих регистров во время, когда GLOBAL_ENABLE = 1, то это не повлияет на работу блока. Чтобы новые значения вступили в силу, требуется пересбросить блок через GLOBAL_ENABLE.

3.16.4.5 Регистр CODER_SEL

Таблица 3.16.7 — Описание регистра CODER_SEL

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
1	2	3
31 – 17	–	
16	HARD_SOFT (RW)	Опция для декодера Витерби. 0 — жесткие решения (ширина шины данных равна 1 бит), 1 — мягкие решения (ширина шины данных равна 3 бита)
15 – 3	–	

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.
Изм	Лист	№ докум.	Подп.	Дата
				Микросхема интегральная 5023BC016
				Техническое описание
				Версия 2.04 от 01.09.2017 г.
				Лист
				221

Продолжение таблицы 3.16.7

1	2	3
2	CONV_SEL (RW)	1 — выбор сверточного кодера Витерби
1	BCH_SEL (RW)	1 — выбор декодера БЧХ, исправляет одну ошибку в 7 байт данных
0	RAND_SEL (RW)	1 — выбор рандомизатора

3.16.4.6 Регистр PHY_FIFO_OVERFLOW

Таблица 3.16.8 — Описание регистра PHY_FIFO_OVERFLOW

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 1	–	
0	PHY_FIFO_OVERFLOW (RW1C)	флаг переполнения памяти асинхронного блока FIFO, когда дескрипторы не готовы к приёму данных, а GLOBAL_ENABLE равен 1, то данные накапливаются в FIFO. При переполнении необходимо сделать пересброс через GLOBAL_ENABLE.

3.16.4.7 Регистр START_SEQ

Таблица 3.16.9 — Описание регистра START_SEQ

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31	ALLOW_1_ERR (RW)	1 — позволить единичную ошибку в маркере начала фрейма. Значение по сбросу: 0
30 – 16	–	
15 – 0	START_SEQ	маркер начала фрейма. Первый принятый бит сравнивается с 15 битом START_SEQ, последний принятый бит сравнивается с 0 битом

3.16.4.8 Регистры TAIL_SEQ1, TAIL_SEQ2

Таблица 3.16.10 — Описание регистров TAIL_SEQ1, TAIL_SEQ2

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 0	TAIL_SEQ (RW)	маркер конца фрейма. Первый принятый бит сравнивается с 31 битом регистра TAIL_SEQ1, последний принятый бит сравнивается с 0 битом TAIL_SEQ2.

3.17 Специализированный вычислитель с плавающей точкой

Блок специализированного вычислителя выполняет операции сложения/вычитания, а также умножения чисел с плавающей запятой. Доступ к специализированному вычислителю осуществляется по шине AMBA через регистры.

Име. № подл.	Подп. и дата.
Взам. инв. №	Инв. № дубл.
Подп. и дата	

Изм	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 222
-----	------	----------	-------	------	---	-------------

Описание регистров специализированного вычислителя приведено в таблице 3.17.1. Описание регистров специализированного вычислителя приведено в таблицах 3.17.2 – 3.17.9.

Таблица 3.17.1 — Описание регистров специализированного вычислителя

Смещение	Название регистра	Тип	Значение по сбросу	Описание
0x00	ADD_SUB	RW	0	Выбор режима сложение/вычитание
0x04	ADD_A	RW	0	Операнд А для операции сложения/вычитания
0x08	ADD_B	RW	0	Операнд В для операции сложения/вычитания
0x0C	MUL_A	RW	0	Первый множитель для операции умножения
0x10	MUL_B	RW	0	Второй множитель для операции умножения
0x14	CONTROL	RW	0	Регистр управления
0x18	ADD_OUT	RO	0	Результат операции сложения
0x1C	MUL_OUT	RO	0	Результат операции умножения
0x20-0xFFFF	—			Не используется, читается 0

3.17.1 Описание регистров

3.17.1.1 Регистр ADD_SUB

Таблица 3.17.2 — Описание регистра ADD_SUB

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 1	—	
0	ADD_SUB	0 — будет выполняться операция вычитания. 1 — будет выполняться операция сложения

3.17.1.2 Регистр ADD_A

Таблица 3.17.3 — Описание регистра ADD_A

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 0	ADD_A	значение первого слагаемого для операции сложения, в формате float.

3.17.1.3 Регистр ADD_B

Таблица 3.17.4 — Описание регистра ADD_B

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 0	ADD_B	значение второго слагаемого для операции сложения, в формате float

Име. № подл.

Подп. и дата.

Взам. инв. №

Инв. № дубл.

3.17.1.1 Регистр ADD_SUB

Таблица 3.17.2 — Описание регистра ADD_SUB

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 1	–	
0	ADD_SUB	0 — будет выполняться операция вычитания. 1 — будет выполняться операция сложения

3.17.1.2 Регистр ADD_A

Таблица 3.17.3 — Описание регистра ADD_A

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 0	ADD_A	значение первого слагаемого для операции сложения, в формате float.

3.17.1.3 Регистр ADD_B

Таблица 3.17.4 — Описание регистра ADD_B

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 0	ADD_B	значение второго слагаемого для операции сложения, в формате float

					Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 223
Изм	Лист	№ докум.	Подп.	Дата		

3.17.1.4 Регистр MUL_A

Таблица 3.17.5 — Описание регистра MUL_A

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 0	MUL_A	значение первого множителя для операции умножения, в формате float

3.17.1.5 Регистр MUL_B

Таблица 3.17.6 — Описание регистра MUL_B

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 0	MUL_B	значение второго множителя для операции умножения, в формате float

3.17.1.6 Регистр CONTROL

Таблица 3.17.7 — Описание регистра CONTROL

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 4	–	
3	MUL_DONE	принимает значение «1» после завершения операции умножения, сбрасывается в «0» после чтения регистра
2	ADD_DONE	принимает значение «1» после завершения операции сложения, сбрасывается в «0» после чтения регистра
1	MUL (W)	запись «1» запускает выполнение операции умножения
0	ADD (W)	запись «1» запускает выполнение операции сложения

3.17.1.7 Регистр ADD_OUT

Таблица 3.17.8 — Описание регистра ADD_OUT

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 0	ADD_OUT (R)	результат операции сложения. Результат доступен через 5 тактов системной частоты после записи в регистр CONTROL

3.17.1.8 Регистр MUL_OUT

Таблица 3.17.9 — Описание регистра MUL_OUT

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 0	MUL_OUT (R)	результат операции умножения. Результат доступен через 4 такта системной частоты после записи в регистр CONTROL

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.

Изм.	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 224
------	------	----------	-------	------	---	-------------

3.18 Контроллер интерфейса CAN

Контроллер интерфейса CAN поддерживает протоколы CAN 20.A и CAN 2.0 B. Структурная схема контроллера изображена на рисунке 3.18.1.

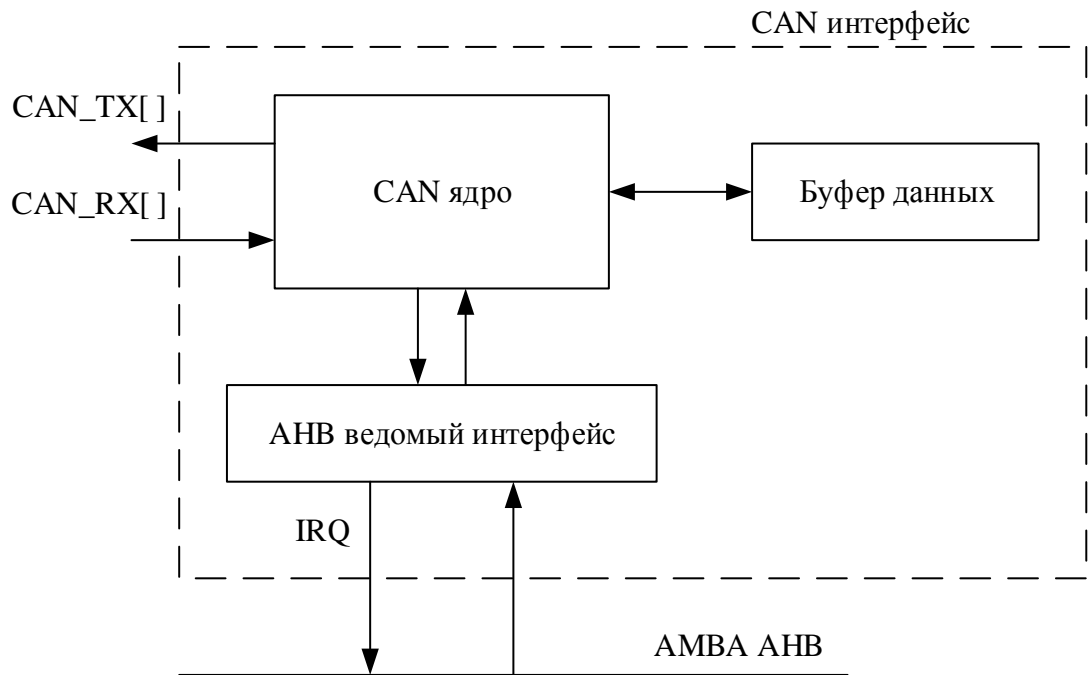


Рисунок 3.18.1 — Структурная схема контроллера интерфейса CAN

3.18.1 Описание

Контроллер может функционировать в базовом и расширенном режимах. Расширенный режим поддерживает расширения протокола CAN 2.0B. Базовый режим имеет более простой интерфейс управления контроллером через регистры, но поддерживает только сообщения стандартного формата Standard Frame Format (SFF). Расширенный режим поддерживает помимо стандартного также и расширенный формат сообщений Extended Frame Format (EFF).

Управление контроллером происходит через шину AMBA посредством обращения к регистрам. Набор регистров различается для базового и расширенного режимов работы. Регистры для обоих режимов расположены в едином адресном пространстве. Все регистры имеют размер 1 байт. Часть регистров контроллера имеет одинаковые функции для базового и расширенного режимов работы, в частности регистр выбора режима работы и регистры временных параметров шины.

Име. № подл.	Подп. и дата.
Взам. инв. №	Инв. № дубл.
Подп. и дата	
Име. № подл.	

					Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист
						225
Изм	Лист	№ докум.	Подп.	Дата		

Как в базовом, так и в расширенном режимах работы контроллер может находиться либо в состоянии сброса, либо в рабочем состоянии. Состояние сброса используется для настройки параметров контроллера, а рабочий режим — для передачи и приёма данных по интерфейсу CAN.

3.18.2 Базовый режим работы

Список регистров для базового режима работы контроллера приведён в таблице 3.18.1. Функциональность регистра, расположенного по определённом адресу, зависит от того, находится контроллер в состоянии сброса либо в рабочем состоянии.

Описание некоторых регистров контроллера CAN приведено в таблицах 3.18.2 – 3.18.5.

Таблица 3.18.1 — Обобщенные данные о регистрах контроллера CAN в базовом режиме

Смещение	Название регистра	Тип	Значение по сбросу	Описание
1	2	3	4	5
0x00	Control	RW	0x21	Контрольный регистр
0x04	Command	WO	0xFF	Регистр команд
0x08	Status	RO	0x0C	Регистр статуса
0x0C	Interrupt	RO	0xE0	Регистр прерываний
0x10	Acceptance_code	RW	0x00	Регистр приёмного кода. Только в состоянии сброса
0x14	Acceptance_mask	RW	0x00	Регистр приёмной маски. Только в состоянии сброса
0x18	Bus_timing_0	RW	0x00	Первый регистр временных параметров шины. Только в состоянии сброса
0x1C	Bus_timing_1	RW	0x00	Второй регистр временных параметров шины. Только в состоянии сброса
0x20	Reserved		0x00	
0x24	Reserved		0x00	
0x28	TX_id1	RW	0xFF	Первый регистр заголовка пакета для передачи. Только в рабочем состоянии
0x2C	TX_id2_rtr_dlc	RW	0xFF	Второй регистр заголовка пакета для передачи. Только в рабочем состоянии

Име. № подл.	Подп. и дата	Взам. име. №	Име. № дубл.	Подп. и дата.

Изм	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 226
-----	------	----------	-------	------	---	-------------

Продолжение таблицы 3.18.1

1	2	3	4	5
0x30	TX_data_byte_1	RW	0xFF	Регистры содержимого пакета для передачи
0x34	TX_data_byte_2	RW	0xFF	
0x38	TX_data_byte_3	RW	0xFF	
0x3C	TX_data_byte_4	RW	0xFF	
0x40	TX_data_byte_5	RW	0xFF	
0x44	TX_data_byte_6	RW	0xFF	
0x48	TX_data_byte_7	RW	0xFF	
0x4C	TX_data_byte_8	RW	0xFF	
0x50	RX_id1	RO	0x00	Первый регистр заголовка пакета для приёма
0x54	RX_id2_rtr_dlc	RO	0x00	Второй регистр заголовка пакета для приёма
0x58	RX_data_byte_1	RO	0x00	Регистры содержимого принимаемого пакета
0x5C	RX_data_byte_2	RO	0x00	
0x60	RX_data_byte_3	RO	0x00	
0x64	RX_data_byte_4	RO	0x00	
0x68	RX_data_byte_5	RO	0x00	
0x6C	RX_data_byte_6	RO	0x00	
0x70	RX_data_byte_7	RO	0x00	
0x74	RX_data_byte_8	RO	0x00	
0x78	Reserved		0x00	
0x7C	Clock_divider	RW	0x00	Регистр делителя рабочей частоты

3.18.2.1 Контрольный регистр Control

Регистр содержит бит управления состоянием контроллера, осуществляющий переход из рабочего состояния в состояние сброса и обратно, и биты включения прерываний.

Таблица 3.18.2 — Описание регистра Control

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
1	2	3
7 – 5	–	Зарезервировано
4	Overrun Interrupt Enable	Включение прерывания переполнения. 1 — включено, 0 — отключено.
3	Error Interrupt Enable	Включение прерывания ошибки. 1 — включено, 0 — отключено.
2	Transmit Interrupt Enable	Включение прерывания передачи. 1 — включено, 0 — отключено.

Име. № подл.	Подп. и дата.
Взам. инв. №	Инв. № дубл.
Подп. и дата	
Име. № подл.	

Изм	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 227
-----	------	----------	-------	------	---	-------------

Продолжение таблицы 3.18.2

1	2	3
1	Receive Interrupt Enable	Включение прерывания приёма. 1 — включено, 0 — отключено.
0	Reset request	Запись «1» в этот бит прерывает любую текущую передачу данных и переводит контроллер в состояние сброса. Запись «0» переводит контроллер в рабочее состояние. По умолчанию активно состояние сброса, бит имеет значение «1».

3.18.2.2 Регистр команд Command

Запись бит данного регистра запускает одно из действий, поддерживаемых ядром контроллера.

Таблица 3.18.3 — Описание регистра Command

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
7 – 4	—	Зарезервировано
3	Clear data overrun	Запись «1» очищает бит переполнения (Data overrun status) в регистре статуса
2	Release receive buffer	Запись «1» очищает приёмный буфер для новых данных
1	Abort transmission	Запись «1» прерывает передачу, которая еще не была запущена
0	Transmission request	Запись «1» запускает передачу сообщения в буфер для отправки (TX buffer)

Передача данных начинается с помощью записи «1» в бит Transmission request. Передача может быть прервана только с помощью записи «1» в бит Receive Interrupt Enable и только если передача еще не была начата. В противном случае она не будет остановлена, но не будет автоматически перезапущена в случае, если произойдет ошибка.

Запись в бит Release receive buffer следует производить после вычитывания содержимого приёмного буфера, чтобы освободить его. Если в FIFO присутствует следующее принятое сообщение, будет сгенерировано новое прерывание приёма (если оно было включено в регистре Control), и бит Receive buffer status в регистре Status будет выставлен повторно.

Име. № подл.	Подп. и дата.	Име. № дубл.	Взам. име. №	Подп. и дата	Изм	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 228
--------------	---------------	--------------	--------------	--------------	-----	------	----------	-------	------	---	-------------

3.18.2.3 Регистр статуса Status

Регистр статуса доступен только на чтение и отражает текущее состояние контроллера.

Таблица 3.18.4 — Описание регистра Status

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
7	Bus status	Значение «1» если контроллер не подключен к шине CAN и не участвует в активности на шине
6	Error Status	Значение «1» когда хотя бы один из счётчиков ошибок превысил значение CPU warning limit = 96
5	Transmit status	Значение «1» во время передачи сообщения контроллером
4	Receive status	Значение «1» во время приёма сообщения контроллером
3	Transmission complete	Значение «1» когда предыдущее сообщение было успешно передано Бит сбрасывается в «0» сразу после начала передачи и не устанавливается в «1» до тех пор, пока сообщение не будет успешно передано.
2	Transmit buffer status	Значение «1» означает, что запись в передающий буфер разрешена Во время передачи сообщения передающий буфер запирается и бит имеет значение «0». Если контроллер не передает сообщения, бит имеет значение «1» и возможен запуск следующих сообщений на передачу.
1	Data overrun status	Значение «1» означает, что сообщение было потеряно из-за отсутствия свободного места в FIFO Бит означает, что полученное контроллером сообщение было утеряно, поскольку в приёмном FIFO отсутствовало свободное место. Данный бит устанавливается в момент вычитывания данных из FIFO.
0	Receive buffer status	Значение «1» означает, что в приёмном FIFO присутствуют принятые сообщения Бит сбрасывается в «0», если была подана команда Release receive buffer через бит в регистре Command. Если при этом в приёмном FIFO присутствуют последующие сообщения, бит будет установлен в «1» повторно.

3.18.2.4 Регистр прерываний Interrupt

Данный регистр содержит информацию о том, что вызвало прерывание в контроллере. Биты устанавливаются, только если соответствующее прерывание было включено в регистре Control.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.					
Изм	Лист	№ докум.	Подп.	Дата					
					Микросхема интегральная 5023BC016				
					Техническое описание				
					Версия 2.04 от 01.09.2017 г.				
					Лист				
					229				

Таблица 3.18.5 — Описание регистра Interrupt

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
7 – 4	–	Зарезервировано
3	Data overrun interrupt	Устанавливается в «1» когда бит Data overrun status в регистре Status переходит из «0» в «1»
2	Error interrupt	Устанавливается значение «1» если изменяется статус шины или статус ошибки
1	Transmit interrupt	Устанавливается значение «1» в момент освобождения передающего буфера
0	Receive interrupt	Устанавливается значение «1», если в приёмном FIFO присутствуют принятые сообщения

Данный регистр сбрасывается после каждого чтения за исключением бита Receive interrupt, который сбрасывается только после передачи команды Release receive buffer записью в регистр Command.

3.18.2.5 Регистры передающего буфера

Содержимое передающего буфера формируется с помощью записи в регистры с адресами от 0x28 до 0x4C, как показано в таблице 3.18.6. Данные регистры формируют сообщение стандартного формата (SFF). Все эти регистры доступны на запись только когда контроллер находится в рабочем состоянии.

Таблица 3.18.6 — Формат регистров, формирующих содержимое передающего буфера

Смещение	Имя регистра	Назначение бит							
		7	6	5	4	3	2	1	0
0x28	TX_id1	ID.10	ID.9	ID.8	ID.7	ID.6	ID.5	ID.4	ID.3
0x2C	TX_id2_rtr_dlc	ID.2	ID.1	ID.0	RTR	DLC.3	DLC.2	DLC.1	DLC.0
0x30	TX_data_byte_1	TX byte 1							
0x34	TX_data_byte_2	TX byte 2							
0x38	TX_data_byte_3	TX byte 3							
0x3C	TX_data_byte_4	TX byte 4							
0x40	TX_data_byte_5	TX byte 5							
0x44	TX_data_byte_6	TX byte 6							
0x48	TX_data_byte_7	TX byte 7							
0x4C	TX_data_byte_8	TX byte 8							

Если бит RTR установлен в «1», ни один байт данных не будет передан в сообщении. При этом бит DLC будет передан и должен быть корректно установлен. Поле DLC может иметь значение большее 8, но при этом в сообщении будет

Изм.

Лист

№ докум.

Подп.

Дата

Име. № подл.

Подп. и дата

Взам. инв. №

Инв. № дубл.

Подп. и дата.

запись только когда контроллер находится в рабочем состоянии.

Таблица 3.18.6 — Формат регистров, формирующих содержимое передающего буфера

Смещение	Имя регистра	Назначение бит							
		7	6	5	4	3	2	1	0
0x28	TX_id1	ID.10	ID.9	ID.8	ID.7	ID.6	ID.5	ID.4	ID.3
0x2C	TX_id2_rtr_dlc	ID.2	ID.1	ID.0	RTR	DLC.3	DLC.2	DLC.1	DLC.0
0x30	TX_data_byte_1	TX byte 1							
0x34	TX_data_byte_2	TX byte 2							
0x38	TX_data_byte_3	TX byte 3							
0x3C	TX_data_byte_4	TX byte 4							
0x40	TX_data_byte_5	TX byte 5							
0x44	TX_data_byte_6	TX byte 6							
0x48	TX_data_byte_7	TX byte 7							
0x4C	TX_data_byte_8	TX byte 8							

Если бит RTR установлен в «1», ни один байт данных не будет передан в сообщении. При этом бит DLC будет передан и должен быть корректно установлен. Поле DLC может иметь значение большее 8, но при этом в сообщении будет

Име. № подл.

Подп. и дата

Взам. инв. №

Инв. № дубл.

Подп. и дата.

Изм.

Лист

№ докум.

Подп.

Дата

Микросхема интегральная 5023BC016

Техническое описание

Версия 2.04 от 01.09.2017 г.

Лист

230

передано не более 8 байт. Устанавливать значение большее 8 не рекомендуется из соображений совместимости со стандартом.

3.18.2.6 Регистры приёмного буфера

Приёмное FIFO имеет размер 64 байта. Значения 10 байт этого FIFO, формирующие наиболее раннее из принятых сообщений, доступны на чтение через регистры с адресами 0x50 – 0x74. Значение бит регистров приёмного буфера идентично по структуре регистрам передающего буфера, показанным в таблице 3.18.6.

3.18.2.7 Фильтрация принимаемых сообщений

Контроллер может производить фильтрацию принимаемых сообщений, используя регистры Acceptance_code и Acceptance_mask. Сообщения могут быть отфильтрованы на основе содержимого их идентификаторов. Старшие 8 из 11 бит идентификатора принятого сообщения сравниваются с содержимым регистра Acceptance_code, если соответствующий бит в регистре Acceptance_mask сброшен в «0». Если в результате все такие биты совпадают, принятое сообщение сохраняется в приёмном FIFO.

3.18.3 Расширенный режим работы

Список регистров для расширенного режима работы контроллера приведён в таблице 3.18.7. Функциональность регистра, расположенного по определённом адресу, зависит от того, находится контроллер в состоянии сброса либо в рабочем состоянии, а также от выбранного формата сообщений (SFF/EFF).

Таблица 3.18.7 — Обобщенные данные о регистрах контроллера CAN в расширенном режиме

Сме- щение	Название регистра	Тип	Значение по сбросу	Описание
1	2	3	4	5
0x00	Mode	RW	0x21	Регистр режима работы
0x04	Command	WO	0xFF	Регистр команд
0x08	Status	RO	0x0C	Регистр статуса
0x0C	Interrupt	RO	0xE0	Регистр прерываний

Изм.	Лист	№ докум.	Подп.	Дата	в «0». Если в результате все такие биты совпадают, принятое сообщение сохраняется в приёмном FIFO.					
					3.18.3 Расширенный режим работы					
					Список регистров для расширенного режима работы контроллера приведён в таблице 3.18.7. Функциональность регистра, расположенного по определённому адресу, зависит от того, находится контроллер в состоянии сброса либо в рабочем состоянии, а также от выбранного формата сообщений (SFF/EFF).					
					Таблица 3.18.7 — Обобщенные данные о регистрах контроллера CAN в расширенном режиме					
					Сме- щение	Название регистра	Тип	Значение по сбросу	Описание	
					1	2	3	4	5	
					0x00	Mode	RW	0x21	Регистр режима работы	
					0x04	Command	WO	0xFF	Регистр команд	
					0x08	Status	RO	0x0C	Регистр статуса	
					0x0C	Interrupt	RO	0xE0	Регистр прерываний	
Изм.	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.					Лист 231

Продолжение таблицы 3.18.7

1	2	3	4	5
0x10	Interrupt_enable	RW	0x00	Регистр включения прерываний
0x14	Reserved		0x00	
0x18	Bus_timing_0	RW	0x00	Первый регистр временных параметров шины. Доступен на запись только в состоянии сброса
0x1C	Bus_timing_1	RW	0x00	Второй регистр временных параметров шины. Доступен на запись только в состоянии сброса
0x20	Reserved		0x00	
0x24	Reserved		0x00	
0x28	Reserved		0xFF	
0x2C	Arbitration_lost_capture	RO	0xFF	Регистр хранения позиции потери арбитражи
0x30	Error_code_capture	RO	0xFF	Регистр кода ошибки
0x34	Error_warning_limit	RW	0xFF	Регистр максимального количества предупреждений. Доступен на запись только в состоянии сброса
0x38	RX_error_counter	RW	0xFF	Счётчик ошибок при приёме. Доступен на запись только в состоянии сброса
0x3C	TX_error_counter	RW	0xFF	Счётчик ошибок при передаче. Доступен на запись только в состоянии сброса
0x40	FI_SFF/FI_EFF/Acceptance_code_0	RW	0xFF	В зависимости от состояния контроллера и формата сообщения данные регистры имеют различное назначение. В состоянии сброса часть регистров задает приёмный код и маску. В рабочем состоянии при записи формируется содержимое сообщения для передачи, при чтении - вычитывается значение принятого сообщения. Назначение регистров различно для стандартного (SFF) и расширенного (EFF) форматов сообщения.
0x44	ID_1/Acceptance_code_1	RW	0xFF	
0x48	ID_2/Acceptance_code_2	RW	0xFF	
0x4C	Data_1/ID_3/Acceptance_code_3	RW	0xFF	
0x50	Data_2/ID_4/Acceptance_mask_0	RW	0x00	
0x54	Data_3/Data_1/Acceptance_mask_1	RW	0x00	
0x58	Data_4/Data_2/Acceptance_mask_2	RW	0x00	

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.

Изм	Лист	№ докум.	Подп.	Дата

Микросхема интегральная 5023BC016
Техническое описание
Версия 2.04 от 01.09.2017 г.

Лист
232

Продолжение таблицы 3.18.7

1	2	3	4	5
0x5C	Data_5/Data_3/Acceptance_mask_3	RW	0x00	
0x60	Data_6/Data_4	RW	0x00	
0x64	Data_7/Data_5	RW	0x00	
0x68	Data_8/Data_6	RW	0x00	
0x6C	FIFO_1/Data_7	RW	0x00	
0x70	FIFO_2/Data_8	RW	0x00	
0x74	RX_message_counter	RO	0x00	Регистр счётчика принимаемых сообщений
0x78	Reserved		0x00	
0x7C	Clock_divider	RW	0x00	Регистр делителя рабочей частоты

3.18.3.1 Регистр режима Mode

Регистр содержит бит управления состоянием контроллера, осуществляющий переход из рабочего состояния в состояние сброса и обратно, а также переход в специфические режимы работы. Назначение бит регистра Mode приведено в таблице 3.18.8.

Таблица 3.18.8 — Описание регистра Mode

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
7 – 4	—	Зарезервировано
3	Acceptance filter mode	Значение «1» — режим одиночной фильтрации, «0» — режим двойной фильтрации
2	Self test mode	Запись «1» переводит контроллер в режим самодиагностики
1	Listen only mode	Запись «1» переводит контроллер в состояние прослушивания шины
0	Reset mode	Запись «1» в этот бит прерывает любую текущую передачу данных и переводит контроллер в состояние сброса. Запись «0» переводит контроллер в рабочее состояние. По умолчанию активно состояние сброса, бит имеет значение «1»

Биты 1 – 3 доступны на запись только в состоянии сброса.

В состоянии прослушивания шины (listen only mode) контроллер не будет посылать подтверждение приёма сообщений. При этом контроллер продолжает реагировать на ошибки на шине.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.					Лист 233
					Изм.	Лист	№ докум.	Подп.	Дата	

В состоянии самодиагностики контроллер может осуществить успешную передачу без получения подтверждения с шины. Для этого в регистре Command необходимо запустить передачу с помощью бита Self reception request. Подключение к шине необходимо и в этом случае, поскольку внутренний loopback не производится.

3.18.3.2 Регистр команд Command

Запись бит данного регистра запускает одно из действий, поддерживаемых ядром контроллера. Описание регистра Command приведено в таблице 3.18.9.

Таблица 3.18.9 — Описание регистра Command

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
7 – 5	–	Зарезервировано
4	Self reception request	Запись «1» запускает передачу и одновременно приём сообщения
3	Clear data overrun	Запись «1» очищает бит переполнения (Data overrun status) в регистре статуса
2	Release receive buffer	Запись «1» очищает приёмный буфер для новых данных
1	Abort transmission	Запись «1» прерывает передачу, которая еще не была запущена
0	Transmission request	Запись «1» запускает передачу сообщения в буфер для отправки (TX buffer)

Передача данных начинается с помощью записи «1» в бит Reset request. Передача может быть прервана только с помощью записи «1» в бит Receive Interrupt Enable и только если передача еще не была начата. В противном случае она не будет остановлена, но не будет автоматически перезапущена в случае, если произойдет ошибка.

Если одновременно записать биты «0» и «1», будет запущена однократная передача, которая не будет перезапущена в случае ошибок.

Запись в бит Release receive buffer следует производить после вычитывания содержимого приёмного буфера, чтобы освободить его. Если в FIFO присутствует следующее принятое сообщение, будет сгенерировано новое прерывание приёма (если оно было включено в регистре Interrupt_enable), и бит Receive buffer status в регистре Status будет выставлен повторно.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>					Лист
Изм.	Лист	№ докум.	Подп.	Дата						234

Запись бита Self reception request в режиме Self test mode позволяет проверить функциональность контроллера в случае отсутствия других устройств на шине. Сообщение будет одновременно передаваться и приниматься, при этом будут одновременно сгенерированы прерывания приёма и передачи.

3.18.3.3 Регистр статуса Status

Регистр статуса в расширенном режиме идентичен по структуре регистру статуса в базовом режиме.

3.18.3.4 Регистр прерываний Interrupt

Данный регистр содержит информацию о том, что вызвало прерывание в контроллере. Биты устанавливаются, только если соответствующее прерывание было включено в регистре Control. Описание регистра Interrupt приведено в таблице 3.18.10.

Таблица 3.18.10 — Описание регистра Interrupt

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
7 – 4	–	Зарезервировано
7	Bus error interrupt	Устанавливается в «1» если на шине обнаружена ошибка
6	Arbitration lost interrupt	Устанавливается в «1» при потере арбитрации контроллером
5	Error passive interrupt	Устанавливается в «1» когда контроллер переходит между состояниями активной и пассивной ошибки
4	–	Зарезервировано
3	Data overrun interrupt	Устанавливается в «1» когда бит Data overrun status в регистре Status переходит из «0» в «1»
2	Error warning interrupt	Устанавливается значение «1» если изменяется статус шины или статус ошибки
1	Transmit interrupt	Устанавливается значение «1» в момент освобождения передающего буфера
0	Receive interrupt	Устанавливается значение «1», если в приёмном FIFO присутствуют принятые сообщения

Данный регистр сбрасывается после каждого чтения за исключением бита Receive interrupt, который сбрасывается только после передачи команды Release receive buffer записью в регистр Command.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>					Лист
Изм.	Лист	№ докум.	Подп.	Дата						235

3.18.3.5 Регистр включения прерываний Interrupt_enable

Данный регистр включает и отключает различные источники прерываний. Если прерывание активно, соответствующий бит в регистре Interrupt может быть выставлен, и будет активироваться сигнал прерывания. Описание регистра Interrupt_enable приведено в таблице 3.18.11.

Таблица 3.18.11 — Описание регистра Interrupt_enable

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
7	Bus error interrupt	Устанавливается в «1» если на шине обнаружена ошибка
6	Arbitration lost interrupt	Устанавливается в «1» при потере арбитрации контроллером
5	Error passive interrupt	Устанавливается в «1» когда контроллер переходит между состояниями активной и пассивной ошибки
4	—	Зарезервировано
3	Data overrun interrupt	Устанавливается в «1» когда бит Data overrun status в регистре Status переходит из «0» в «1»
2	Error warning interrupt	Устанавливается значение «1» если изменяется статус шины или статус ошибки
1	Transmit interrupt	Устанавливается значение «1» в момент освобождения передающего буфера
0	Receive interrupt	Устанавливается значение «1», если в приёмном FIFO присутствуют принятые сообщения

3.18.3.6 Регистр хранения позиции потери арбитрации Arbitration_lost_capture

Когда контроллер теряет арбитрацию, данный регистр захватывает битовую позицию процессора битового потока. Регистр не будет изменять значение до тех пор, пока он не будет вычитан. Описание регистра Arbitration_lost_capture приведено в таблице 3.18.12.

Таблица 3.18.12 — Описание регистра Arbitration_lost_capture

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
7 – 5	—	Зарезервировано
4 – 0	Bit number	Бит, в котором была потеряна арбитрация

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	Микросхема интегральная 5023BC016					Лист
					Техническое описание					236
					Версия 2.04 от 01.09.2017 г.					
					Изм	Лист	№ докум.	Подп.	Дата	

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.

Таблица 3.18.13 — Описание регистра Error code capture

3.18.3.8 Регистр	максимального	количества	предупреждений
Error_warning_limit			

3.18.3.9 Регистр счётчика ошибок при приёме RX error counter

Регистр показывает значение счётчика ошибок на приёме. В состоянии сброса доступен на запись. Отключение шины сбрасывает этот счётчик в «0».

					Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист
Изм.	Лист	№ докум.	Подп.	Дата		237

Таблица 3.18.15 — Описание регистра TX frame information

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
7	FF	Задаёт формат передаваемого фрейма. 1 — Extended Frame Format (EFF), 0 — Standard Frame Format (SFF)
6	RTR	Необходимо установить в 1 для передачи фрейма, содержащего запрос на удаленную передачу
5	—	
4	—	
3	DLC.3	Поле определяет длину передаваемого сообщения. Должно принимать значения от 0 до 8. Если поле имеет значение большее 8, будет передано 8 Б
2	DLC.2	
1	DLC.1	
0	DLC.0	

Регистры, задающие поле идентификатора фрейма, различаются для стандартного и расширенного форматов фрейма. Регистры для стандартного фрейма описаны в таблице 3.18.16, для расширенного — в таблице 3.18.17.

Таблица 3.18.16 — Описание регистров идентификатора для стандартного формата фрейма (SFF)

Адрес	Регистр	7	6	5	4	3	2	1	0
0x44	TX ID 1	ID.28	ID.27	ID.26	ID.25	ID.24	ID.23	ID.22	ID.21
0x48	TX ID 2	ID.20	ID.19	ID.18	—	—	—	—	—

Таблица 3.18.17 — Описание регистров идентификатора для расширенного формата фрейма (EFF)

Адрес	Регистр	7	6	5	4	3	2	1	0
0x44	TX ID 1	ID.28	ID.27	ID.26	ID.25	ID.24	ID.23	ID.22	ID.21
0x48	TX ID 2	ID.20	ID.19	ID.18	ID.17	ID.16	ID.15	ID.14	ID.13
0x4C	TX ID 3	ID.12	ID.11	ID.10	ID.9	ID.8	ID.7	ID.6	ID.5
0x50	TX ID 4	ID.4	ID.3	ID.2	ID.1	ID.0	—	—	—

Содержимое полей данных для стандартного фрейма задается регистрами по адресам 0x4C – 0x68 и для расширенного фрейма — по адресам 0x54 – 0x70. Данные передаются, начиная с наиболее значимого бита (MSB) по младшему адресу.

3.18.3.12 Регистры приёмного буфера

Содержимое приёмного буфера можно прочитать из регистров по адресам от 0x40 до 0x70, как показано в таблице 3.18.18. Расположение регистров зависит от формата передаваемого сообщения (SFF/EFF). Все эти регистры доступны на чтение

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.

Изм	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 239
-----	------	----------	-------	------	---	-------------

только когда контроллер находится в рабочем состоянии. В состоянии сброса по данным адресам расположены регистры приёмного фильтра сообщений.

Таблица 3.18.18 — Регистры, отображающие содержимое приёмного буфера в стандартном и расширенном сообщениях

Смещение	SFF	EFF
0x40	RX frame information	RX frame information
0x44	RX ID 1	RX ID 1
0x48	RX ID 2	RX ID 2
0x4C	RX data 1	RX ID 3
0x50	RX data 2	RX ID 4
0x54	RX data 3	RX data 1
0x58	RX data 4	RX data 2
0x5C	RX data 5	RX data 3
0x60	RX data 6	RX data 4
0x64	RX data 7	RX data 5
0x68	RX data 8	RX data 6
0x6C	RX FI следующего сообщения в FIFO	RX data 7
0x70	RX ID1 следующего сообщения в FIFO	RX data 8

Регистр формата фрейма (RX frame information), представленный в таблице 3.18.19, имеет одинаковую структуру для стандартных и расширенных фреймов.

Таблица 3.18.19 — Описание регистра RX frame information

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
7	FF	Формат принимаемого фрейма. 1 — Extended Frame Format (EFF), 0 — Standard Frame Format (SFF)
6	RTR	Имеет значение «1», если принятый фрейм имеет формат RTR
5	—	Содержит количество байт данных в принятом фрейме
4	—	
3	DLC.3	
2	DLC.2	
1	DLC.1	
0	DLC.0	

Регистры, содержащие поле идентификатора фрейма, различаются для стандартного и расширенного форматов фрейма. Регистры для стандартного фрейма описаны в таблице 3.18.20, для расширенного — в таблице Таблица 3.18.21.

Изм	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 240
Изм	Лист	№ докум.	Подп.	Дата		

Таблица 3.18.20 — Описание регистров идентификатора для стандартного формата фрейма (SFF)

Адр.	Регистр	7	6	5	4	3	2	1	0
0x44	TX ID 1	ID.28	ID.27	ID.26	ID.25	ID.24	ID.23	ID.22	ID.21
0x48	TX ID 2	ID.20	ID.19	ID.18	RTR	—	—	—	—

Таблица 3.18.21 — Описание регистров идентификатора для расширенного формата фрейма (EFF)

Адр.	Регистр	7	6	5	4	3	2	1	0
0x44	TX ID 1	ID.28	ID.27	ID.26	ID.25	ID.24	ID.23	ID.22	ID.21
0x48	TX ID 2	ID.20	ID.19	ID.18	ID.17	ID.16	ID.15	ID.14	ID.13
0x4C	TX ID 3	ID.12	ID.11	ID.10	ID.9	ID.8	ID.7	ID.6	ID.5
0x50	TX ID 4	ID.4	ID.3	ID.2	ID.1	ID.0	RTR	—	—

Содержимое полей данных для стандартного фрейма содержится в регистрах по адресам 0x4C – 0x68 и для расширенного фрейма — по адресам 0x54 – 0x70.

3.18.3.13 Фильтрация принимаемых сообщений

Фильтр принимаемых сообщений используется, чтобы отфильтровывать сообщения, не соответствующие определённым условиям. Если сообщение отфильтровано, оно не будет помещаться в приёмное FIFO и будет недоступно через регистры.

В расширенном режиме работы существует два различных режима фильтрации: одиночная и двойная фильтрация. Режим фильтрации определяется битом Acceptance filter mode в регистре Mode. В одиночном режиме используется один фильтр размером 4 байта. В двойном режиме используются два фильтра меньшего размера, и, если хотя бы один из этих фильтров не отсеивает сообщение, оно помещается в приёмное FIFO.

Каждый фильтр состоит из двух частей — приёмный код и приёмная маска. Приёмный код содержит шаблон, с которым сравниваются идентификаторы принимаемых сообщений. Приёмная маска определяет биты, значение которых не имеет значения.

Ине. № подл.	Подп. и дата.
Взам. инв. №	Инв. № дубл.
Подп. и дата	
Ине. № подл.	

Изм.	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 241
------	------	----------	-------	------	---	-------------

Всего приёмный фильтр использует 8 регистров, расположенных по адресам 0x40 – 0x5C, и доступных только в состоянии сброса (см. таблицу 3.18.22).

Таблица 3.18.22 — Регистры фильтра принимаемых сообщений

Смещение	Регистр
0x40	Acceptance code 0 (ACR0)
0x44	Acceptance code 1 (ACR1)
0x48	Acceptance code 2 (ACR2)
0x4C	Acceptance code 3 (ACR3)
0x50	Acceptance mask 0 (AMR0)
0x54	Acceptance mask 1 (AMR1)
0x58	Acceptance mask 2 (AMR2)
0x5C	Acceptance mask 3 (AMR3)

3.18.3.13.1 Одиночный режим фильтрации, стандартные фреймы

Регистры ACR0-3 сравниваются с содержимым входящего сообщения в следующем порядке:

- ACR0.7-0 & ACR1.7-5 сравниваются с ID.28-18;
- ACR1.4 сравнивается с битом RTR;
- ACR1.3-0 не используются;
- ACR2 & ACR3 сравниваются с байтами данных 1 и 2.

Соответствующие биты в регистрах AMR, выставленные в «1», означают, что значения данных бит могут быть любыми для приёма сообщения.

3.18.3.13.2 Одиночный режим фильтрации, расширенные фреймы

Регистры ACR0-3 сравниваются с содержимым входящего сообщения в следующем порядке:

- ACR0.7-0 & ACR1.7-0 сравниваются с ID.28-13;
- ACR2.7-0 сравнивается с ID.12-0;
- ACR3.2 сравнивается с битом RTR;
- ACR3.1-0 не используются.

Соответствующие биты в регистрах AMR, выставленные в «1», означают, что значения данных бит могут быть любыми для приёма сообщения.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	Микросхема интегральная 5023BC016					Лист
					Техническое описание					242
					Версия 2.04 от 01.09.2017 г.					
					Изм	Лист	№ докум.	Подп.	Дата	

3.18.3.13.3 Двойной режим фильтрации, стандартные фреймы

Регистры ACR0-3 сравниваются с содержимым входящего сообщения в следующем порядке:

Фильтр 1:

- ACR0.7-0 & ACR1.7-5 сравниваются с ID.28-18;
- ACR1.4 сравнивается с битом RTR;
- ACR1.3-0 сравниваются со старшим полубайтом байта данных 1;
- ACR3.3-0 сравниваются с младшим полубайтом байта данных 1;

Фильтр 2:

- ACR2.7-0 & ACR3.7-5 сравниваются с ID.28-18;
- ACR3.4 сравниваются с битом RTR.

Соответствующие биты в регистрах AMR, выставленные в «1», означают, что значения данных бит могут быть любыми для приёма сообщения.

3.18.3.13.4 Двойной режим фильтрации, расширенные фреймы

Регистры ACR0-3 сравниваются с содержимым входящего сообщения в следующем порядке:

Фильтр 1:

- ACR0.7-0 & ACR1.7-0 сравниваются с ID.28-13.

Фильтр 2:

- ACR2.7-0 & ACR3.7-0 сравниваются с ID.28-13.

Соответствующие биты в регистрах AMR, выставленные в «1», означают, что значения данных бит могут быть любыми для приёма сообщения.

3.18.3.14 Регистр счётчика принимаемых сообщений RX message counter

Регистр содержит количество сообщений, которые в настоящий момент находятся в приёмном FIFO. Старшие 3 бита регистра всегда имеют значение «0».

3.18.4 Общие регистры для базового и расширенного режимов работы

Три регистра имеют одинаковое расположение и функциональность как в базовом, так и в расширенном режимах работы. Это регистры делителя частоты и временных параметров шины.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата	<div>Микросхема интегральная 5023BC016</div> <div>Техническое описание</div> <div>Версия 2.04 от 01.09.2017 г.</div>					Лист 243	
Изм.	Лист	№ докум.	Подп.	Дата							

3.18.4.1 Регистр делителя частоты Clock_divider

Данный регистр в текущей конфигурации контроллера используется только для выбора режима работы — базового или расширенного. Поля, задающие частоту, управляют выводом clkout, который ни к чему не подключен. Описание полей регистра Clock_divider приводится в таблице 3.18.23.

Таблица 3.18.23 — Описание регистра Clock_divider

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
7	CAN mode	1 — расширенный режим, 0 — базовый режим
6 – 4	–	Зарезервировано
3	Clock off	Отключение вывода clkout
2 – 0	Clock divisor	Выбор частоты (не используется)

3.18.4.2 Первый регистр временных параметров Bus_timing_0

Описание полей регистра Bus_timing_0 приводится в таблице 3.18.24.

Таблица 3.18.24 — Описание регистра Bus_timing_0

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
7 – 6	SJW	Synchronization jump width
5 – 0	BRP	Предделитель Baud rate

Период тактового сигнала контроллера CAN определяется по формуле

$$tscl = 2 \cdot tclk \cdot (BRP + 1), \quad (12)$$

где tclk — период системной частоты.

Величина SJW определяет, сколько периодов tscl необходимо для одной пересинхронизации.

3.18.4.3 Второй регистр временных параметров Bus_timing_1

Описание полей регистра Bus_timing_1 приводится в таблице 3.18.25.

Таблица 3.18.25 — Описание регистра Bus_timing_1

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
7	SAM	1 — шина сэмплируется 3 раза, 0 — единичное сэмплирование
6 – 4	TSEG2	Time segment 2
3 – 0	TSEG1	Time segment 1

Име. № подл.	Подп. и дата.	Име. № дубл.	Взам. име. №	Име. № подл.	Подп. и дата.	Изм	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 244
--------------	---------------	--------------	--------------	--------------	---------------	-----	------	----------	-------	------	---	-------------

Длительность битового периода на шине CAN определяется системной частотой CAN и временными сегментами Time segment 1 и 2:

$$ttseg1 = tscl \cdot (TSEG1 + 1) \tag{13}$$

$$ttseg2 = tscl \cdot (TSEG2 + 1) \tag{14}$$

$$tbit = ttseg1 + ttseg2 + tscl \tag{15}$$

Сэмплирование производится между TSEG1 и TSEG2 в битовом периоде.

3.19 Контроллер интерфейса I²C

Контроллер интерфейса I²C реализует функции мастера на шине. Блок I²C-мастер расположен на шине AMBA APB. Он является совместимым со стандартом Philips I²C. Поддерживается режим скорости Standard (100 кбит/с) и режим Fast (400 кбит/с). На обеих линиях шины I²C должны присутствовать внешние резисторы, подтягивающие линию к высокому уровню. Структурная схема блока представлена на рисунке 3.19.1.

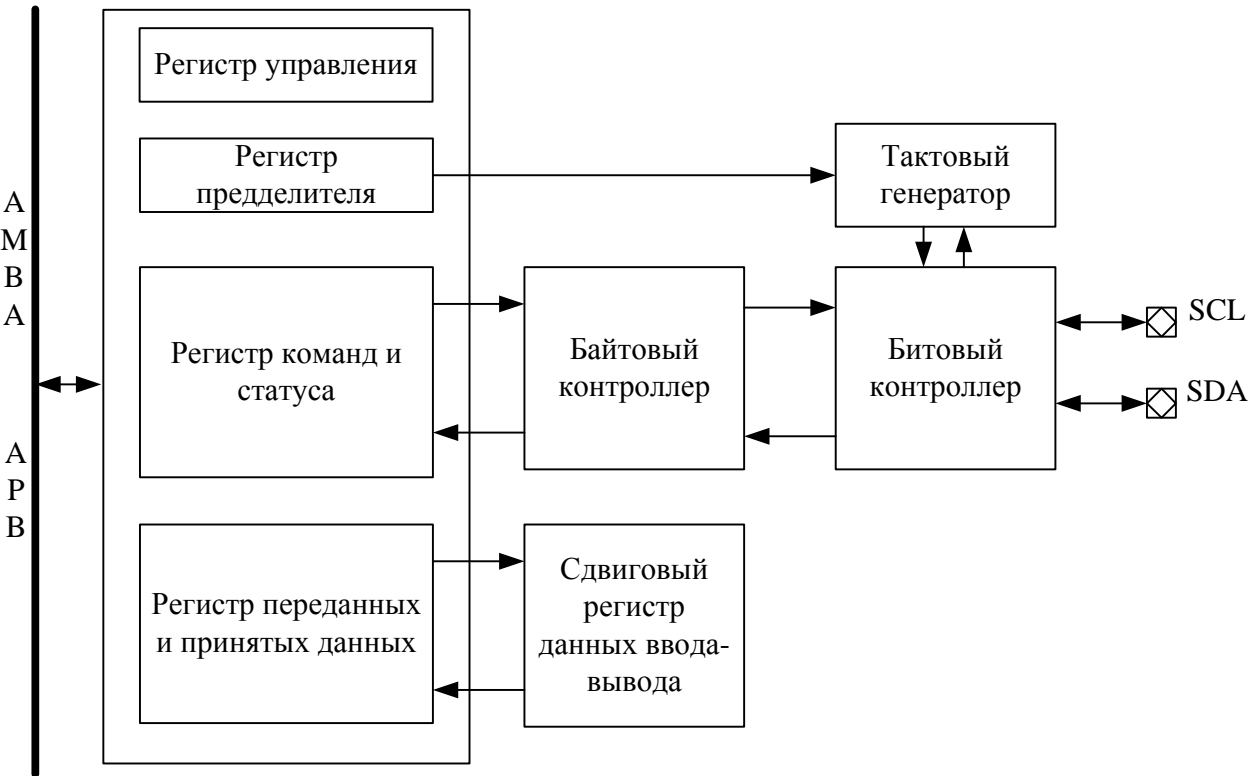


Рисунок 3.19.1 — Структурная схема блока I²C-мастер

3.19.1 Описание работы блока I²C-мастер

3.19.1.1 Протокол передачи

Шина I²C является простой 2-проводной последовательной шиной с поддержкой нескольких мастеров с поддержкой детектирования коллизий и арбитражей.

Шина состоит из последовательной линии данных (SDA) и последовательной линии тактового сигнала (SCL). Стандартом I²C определяются 3 режима скорости: Standard (100 кбит/с), Fast (400 кбит/с) и High speed (3,4 Мбит/с), из них блоком поддерживаются первые два.

Транзакция на шине I²C начинается со стартового условия (START). Условие START определяется как переход с высокого на низкий уровень на линии SDA в момент, когда на линии SCL высокий уровень. Транзакция оканчивается условием STOP, определяемым как переход с низкого на высокий уровень на линии SDA в момент, когда на линии SCL высокий уровень. Эти условия всегда генерируются мастером на шине. Шина считается занятой после условия START и свободной после определённого количества времени после условия STOP. Время, между условием STOP и последующим условием START определяется в спецификации I²C и зависит от скорости шины.

На рисунке 3.19.2 изображена транзакция записи данных на шине I²C. Сначала мастер генерирует условие START и затем передаёт 7-битный адрес подчинённого устройства (slave). За адресом следует R/\bar{W} бит, определяющий направление передачи. В случае, изображённом на рисунке 3.19.2, бит R/\bar{W} имеет значение «0», что соответствует операции записи. После того, как мастер передаёт адрес и бит R/\bar{W} , он освобождает линию SDA. Приёмник подтягивает линию SDA к низкому уровню для подтверждения передачи. Если приёмник не подтверждает передачу, мастер может сгенерировать условие STOP для того, чтобы прервать транзакцию, либо начать новую транзакцию сгенерировав повторное условие START.

После того, как первый байт был подтверждён приёмником, мастер передаёт байт данных. Если бит R/\bar{W} был установлен в «1», мастер должен был действовать как приёмник в течение этой фазы транзакции. После того, как байт данных был

Име. № подл.	Подп. и дата	Взам. име. №	Име. № дубл.	Подп. и дата.	<div>Микросхема интегральная 5023BC016</div> <div>Техническое описание</div> <div>Версия 2.04 от 01.09.2017 г.</div>					Лист
Изм.	Лист	№ докум.	Подп.	Дата						246

передан, приёмник подтверждает приём байта и мастер генерирует условие STOP для того, чтобы завершить транзакцию.

Если скорость на шине слишком велика для подчинённого устройства, оно может растянуть период тактового сигнала, удерживая сигнал на линии SCL на низком уровне после того, как мастер установил на линии значение SCL в низкий уровень.

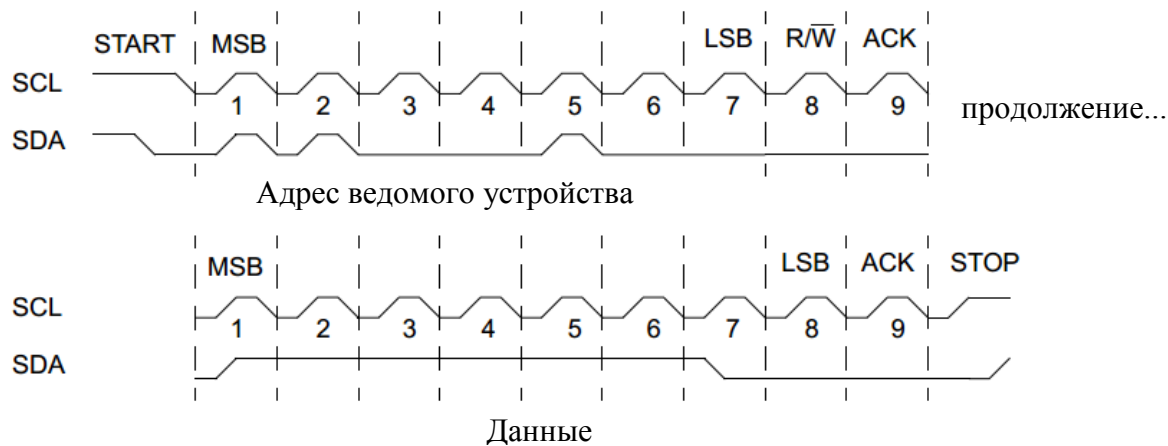


Рисунок 3.19.2 — Полная транзакция на шине I²C

3.19.1.2 Генерация тактового сигнала

Блок I²C-мастер использует регистр предделителя частоты на линии SCL, содержащий значение Prescale. Значение предделителя рассчитывается по формуле

$$Prescale = FSYS / (5 \cdot FSCL) - 1, \tag{16}$$

где FSCL имеет значение 100 кГц для режима скорости Standard и 400 кГц для режима Fast.

То есть для системной частоты 80 МГц значение предделителя будет равно

$$Prescale = 80 \text{ МГц} / (5 \cdot 100 \text{ кГц}) - 1 = 159 = 0x9F. \tag{17}$$

Значение предделителя частоты может быть изменено только когда блок отключен. В связи с особенностями синхронизации минимальное рекомендуемое значение делителя равно трём. Таким образом минимальная системная частота для функционирования в режиме скорости Standard равна 2 МГц. При этом системная частота, равная 2 МГц, приведёт к несоответствию требованиям на минимальное время setup, равное 100 нс в режиме скорости Fast.

Име. № подл.	Подп. и дата	Блок I ² C-мастер использует регистр предделителя частоты на линии SCL, содержащий значение Prescale. Значение предделителя рассчитывается по формуле									
		$Prescale = FSYS / (5 \cdot FSCL) - 1,$									
		где FSCL имеет значение 100 кГц для режима скорости Standard и 400 кГц для режима Fast.									
		То есть для системной частоты 80 МГц значение предделителя будет равно									
Име. № дубл.	Подп. и дата	$Prescale = 80 \text{ МГц} / (5 \cdot 100 \text{ кГц}) - 1 = 159 = 0x9F.$									
		Значение предделителя частоты может быть изменено только когда блок отключен. В связи с особенностями синхронизации минимальное рекомендуемое значение делителя равно трём. Таким образом минимальная системная частота для функционирования в режиме скорости Standard равна 2 МГц. При этом системная частота, равная 2 МГц, приведёт к несоответствию требованиям на минимальное время setup, равное 100 нс в режиме скорости Fast.									
Взам. инв. №	Подп. и дата										
Име. № инв.	Подп. и дата										
Изм.		Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016					Лист
						Техническое описание					
						Версия 2.04 от 01.09.2017 г.					
											247

Для совместимости со спецификацией I²C минимальная допустимая системная частота равна 20 МГц. В случае использования системной частоты меньшего значения, необходимо учитывать, что данные от подчинённых устройств должны стабилизироваться на линии SDA за 1 период системной частоты до перехода сигнала на линии SCL в высокий уровень.

3.19.1.3 Программная модель управления блоком

Блок I²C-мастер инициализируется посредством записи значения Prescale в регистр предделителя и дальнейшего включения блока в работу путём установки бита EN в регистре Control. Прерывания включаются путём записи бита IEN в регистре Control.

Чтобы записать байт данных в подчинённое устройство, мастер должен сгенерировать условие START и послать адрес устройства со значением бита R/\overline{W} равным 0. После того, как подчинённое устройство подтверждает адрес, мастер передает данные, ожидает подтверждения и генерирует условие STOP.

Последовательность действий пользователя, необходимая для записи байта данных, следующая:

- 1) выполнить битовый сдвиг адреса подчинённого устройства на одну позицию влево, записать полученное значение в регистр Tx_Rx. Младший бит регистра, соответствующий биту R/\overline{W} , установлен в «0»;
- 2) сгенерировать условие START и послать содержимое регистра Tx_Rx, установив биты STA_RxACK и WR в регистр Com_Stat;
- 3) дождаться прерывания, либо момента времени, когда бит TIP в регистре Com_Stat станет равным нулю;
- 4) прочесть бит STA_RxACK в регистре Com_Stat. Если STA_RxACK в нуле, это значит, что подчинённое устройство подтвердило транзакцию и можно перейти к п. 5. Если же STA_RxACK имеет значение «1», подчинённое устройство не подтвердило транзакцию, нужно перейти к п. 1;
- 5) записать значение передаваемого байта данных в регистр Tx_Rx;

Име. № подл.	Подп. и дата	Взам. име. №	Име. № дубл.	Подп. и дата.	<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>					Лист
Изм.	Лист	№ докум.	Подп.	Дата						248

6) послать данные в подчинённое устройство и сгенерировать условие STOP посредством установки битов STO_BUSY и WR в регистр Com_Stat;

7) дожждаться прерывания, либо момента времени, когда бит TIP в регистре Com_Stat станет равным нулю;

8) убедиться, что подчиненное устройство подтвердило данные, прочитав бит STA_RxACK в регистре Com_Stat. STA_RxACK должен иметь значение «0».

Чтобы прочитать байт данных из внешней памяти, работающей по интерфейсу I²C, большая часть приведённой выше последовательности действий повторяется. Записываемые данные в этом случае являются адресом в памяти. После того, как адрес был записан, мастер генерирует условие START и читает данные из подчинённого устройства.

Последовательность действий, которую нужно осуществить, чтобы прочитать из внешней I²C-памяти:

1) выполнить битовый сдвиг адреса подчинённого устройства на одну позицию влево, записать полученное значение в регистр Tx_Rx. Младший бит регистра, соответствующий биту R/\overline{W} , установлен в «0»;

2) сгенерировать условие START и послать содержимое регистра Tx_Rx, установив биты STA_RxACK и WR в регистр Com_Stat;

3) дожждаться прерывания, либо момента времени, когда бит TIP в регистре Com_Stat станет равным нулю;

4) прочитать бит STA_RxACK в регистре Com_Stat. Если STA_RxACK в нуле, это значит, что подчинённое устройство подтвердило транзакцию и можно перейти к п. 5. Если же STA_RxACK имеет значение «1», подчинённое устройство не подтвердило транзакцию, нужно перейти к п. 1;

5) записать адрес в устройстве, который нужно будет прочитать из памяти, в регистр Tx_Rx;

6) установить бит WR в регистр Com_Stat. Условие STOP не должно генерироваться;

7) дожждаться прерывания, либо момента времени, когда бит TIP в регистре Com_Stat станет равным нулю;

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.						Лист
Изм.	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016					249
					Техническое описание					
					Версия 2.04 от 01.09.2017 г.					

8) прочитать бит STA_RxACK в регистре Com_Stat. Значение бита должно быть «0»;

9) снова выполнить битовый сдвиг адреса подчинённого устройства на одну позицию влево, записать полученное значение в регистр Tx_Rx. Младший бит регистра, соответствующий биту R/\overline{W} , установить в «1» (признак транзакции чтения);

10) установить биты STA_RxACK и WR в регистр Com_Stat для того, чтобы сгенерировать повторное условие START;

11) дождаться прерывания, либо момента времени, когда бит TIP в регистре Com_Stat станет равным нулю;

12) прочитать бит STA_RxACK в регистре Com_Stat. Подчинённое устройство должно подтвердить транзакцию (значение бита «0»);

13) подготовиться к приёму данных из I²C-памяти; Установить биты RD_AL, ACK и STO_BUSY в регистре Com_Stat. Установка бита ACK приводит к условию NACK для принятых данных, что означает окончание транзакции;

14) дождаться прерывания, либо момента времени, когда бит TIP в регистре Com_Stat станет равным нулю;

15) принятые данные могут быть прочитаны из регистра Tx_Rx.

Чтобы выполнить последовательное чтение нескольких байт, мастер должен продвигаться между п. 13 и п. 15 не устанавливая бит ACK и STO_BUSY в п. 13. Для того, чтобы завершить последовательный приём данных, эти биты необходимо установить в «1». Не все подчинённые устройства могут поддерживать режим последовательного чтения.

Чтобы записать байт данных в устройство, требующее адресации (например, I²C-память), сначала мастер адресует устройство и указывает адрес для записываемых данных. После того, как подчинённое устройство подтверждает первую часть транзакции, записываемые данные передаются без повторной генерации условия START.

Необходимо выполнить нижеприведённую последовательность действий:

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	<div>Микросхема интегральная 5023BC016</div> <div>Техническое описание</div> <div>Версия 2.04 от 01.09.2017 г.</div>					Лист
Изм.	Лист	№ докум.	Подп.	Дата						250

1) выполнить битовый сдвиг адреса подчинённого устройства на одну позицию влево, записать полученное значение в регистр Tx_Rx. Младший бит регистра, соответствующий биту R/\overline{W} , установлен в «0»;

2) сгенерировать условие START и послать содержимое регистра Tx_Rx, установив биты STA_RxACK и WR в регистр Com_Stat;

3) дождаться прерывания, либо момента времени, когда бит TIP в регистре Com_Stat станет равным нулю;

4) прочитать бит STA_RxACK в регистре Com_Stat. Если STA_RxACK в нуле, это значит, что подчинённое устройство подтвердило транзакцию и можно перейти к п. 5. Если же STA_RxACK имеет значение «1», подчинённое устройство не подтвердило транзакцию, нужно перейти к п. 1;

5) записать адрес в устройства, который нужно будет прочитать из памяти, в регистр Tx_Rx;

6) установить бит WR в регистр Com_Stat;

7) дождаться прерывания, либо момента времени, когда бит TIP в регистре Com_Stat станет равным нулю;

8) прочитать бит STA_RxACK в регистре Com_Stat. Значение бита должно быть «0»;

9) записать байт данных в регистр Tx_Rx;

10) установить биты WR и STO_BUSY в регистр Com_Stat для того, чтобы послать байт данных и затем сгенерировать условие STOP;

11) дождаться прерывания, либо момента времени, когда бит TIP в регистре Com_Stat станет равным нулю;

12) проверить бит STA_RxACK в регистре Com_Stat. Если запись прошла успешно, подчинённое устройство должно подтвердить транзакцию.

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.						
Изм	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016					Лист
					Техническое описание					
					Версия 2.04 от 01.09.2017 г.					251

3.19.2 Описание регистров блока I²C-мастер

Таблица 3.19.1 — Структура регистров блока I²C-мастер

Смещение	Название регистра	Тип	Значение по сбросу	Описание
0x00	Clock_prescale	RW	0	Регистр делителя частоты
0x04	Control	RW	0	Регистр управления
0x08	Tx_Rx	RW	0	Регистр переданных и принятых данных
0x0C	Com_Stat	RW	0	Регистр команд и статуса

3.19.2.1 Регистр Clock_prescale

Таблица 3.19.2 — Описание регистра Clock_prescale

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 16	–	
15 – 0	Prescale (RW)	Значение по сбросу 0xFFFF. Значение используется для задания периода тактового сигнала на линии SCL. Значение может быть изменено только если бит EN в регистре Control установлен в «0». Минимальное рекомендуемое значение этого поля равно 3

3.19.2.2 Регистр Control

Таблица 3.19.3 — Описание регистра Control

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 8	–	
7	EN	1 — Перевод блока I ² C в рабочий режим
6	IEN	1 — Включение прерывания, происходящего по завершению транзакции
5 – 0	–	

3.19.2.3 Регистр Tx_Rx

Таблица 3.19.4 — Описание регистра Tx_Rx

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 8	–	
7 – 0	DATA	Данные для передачи и прёма. По записи: старшие 7 бит слова данных, которые будут переданы по шине I ² C. При этом значение первого бита R/W определяет направление транзакции: 1 — чтение из подчинённого устройства, 0 — запись в подчинённое устройство. По чтению: последний байт, принятый по шине I ² C

Име. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.	<p>Микросхема интегральная 5023BC016</p> <p>Техническое описание</p> <p>Версия 2.04 от 01.09.2017 г.</p>					Лист
Изм	Лист	№ докум.	Подп.	Дата						252

3.19.2.4 Регистр Com_Stat

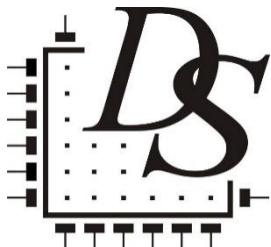
Таблица 3.19.5 — Описание регистра Com_Stat

№ бита	Наименование бита	Функциональное назначение бита, краткое описание бита назначения и принимаемых значений
31 – 8	–	
7	STA_ RxAck	По записи: генерация условия START на шине, либо повторного условия START По чтению: принятое значение бита подтверждения (acknowledge). 0 — подчинённое устройство подтвердило транзакцию, 1 — подтверждение не было получено
6	STO_ BUSY	По записи: генерация условия STOP на шине По чтению: бит выставляется в «1» если на шине было обнаружено условие START и сбрасывается в «0» если было обнаружено условие STOP
5	RD_ AL	По записи: чтение из подчинённого устройства По чтению: устанавливается в «1», когда блок потерял арбитрацию. Данное условие наступает в случае, если обнаружено условие STOP, которое не было запрошено, либо если мастер выставляет высокий уровень на линии SDA, но уровень на линии SDA остается в низком состоянии
4	WR	Запись в подчинённое устройство
3	ACK	Используется, когда мастер является приёмником. 0 — послать ACK, 1 — послать NACK
2	–	
1	TIP	Имеет значение «1», когда происходит передача данных и «0» — когда транзакция завершается. Также этот бит выставляется, когда блок генерирует условие STOP
0	IACK_ IF	По записи: очищает флаг прерывания (IF) в регистре Com_Stat По чтению: бит выставляется, когда транзакция завершена, либо, когда теряется арбитрация. Если установлен бит IEN в регистре Control, в этот момент будет сгенерировано прерывание. Новые прерывания также будут генерироваться даже если этот бит не был сброшен

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.

					Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист
Изм.	Лист	№ докум.	Подп.	Дата		253

По вопросам поставки микросхем и технической поддержки обращайтесь в
ООО «НПП «Цифровые решения»:



ООО «НПП «Цифровые решения»

111250, г. Москва, ул. 2-я Синичкина, д.9а, стр. 7
Почтовый адрес: 105066 Россия, г. Москва, а/я 18
Телефон: +7 (495) 978-2870
Факс: +7 (495) 745-4218
e-mail: support@dsol.ru
www.dsol.ru

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.						
Изм	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.					Лист
										254

Перечень сокращений и условных обозначений

БВС	—	Блок вставки синхромаркера
БПС	—	Блок поиска синхромаркера
ГС	—	Граничное сканирование
ВС	—	Специализированный вычислитель с плавающей запятой
ВСО	—	Внутрикристальная схема отладки
ВСФ	—	Воздействие специальных факторов
КС	—	Командное слово
КВП	—	Контроллер внешней памяти
КШ	—	Контроллер шины
М	—	Режим монитора
МКПД	—	Мультиплексный канал передачи данных
ОЗУ	—	Оперативное запоминающие устройство
ОС	—	Ответное слово
ОУ	—	Оконечное устройство
ПДП	—	Прямой доступ к памяти
ПдТМИ	—	Передачик телеметрической информации
ПрТКИ	—	Приёмник телекомандной информации
СТ	—	Сторожевой таймер
ФК	—	Функциональный контроль
ФТС	—	Формирователь тактовых сигналов и сигналов сброса
ARM	—	Advanced RISK Machine
AMBA AHB	—	Advanced Microcontroller Bus Architecture, Advanced High performance Bus
AMBA APB	—	Advanced Microcontroller Bus Architecture, Advanced Peripheral Bus
AHB	—	Advanced High performance Bus
BusMUX	—	Advanced High performance Bus - Advanced Peripheral Bus
CCSDS	—	Consultative Committee for Space Data Systems
CAN	—	Controller Area Network
CLK	—	Тактовый сигнал системной частоты
CPSDVSR	—	clockprescaledivisorvalue – коэффициент деления тактового сигнала
CPU	—	ЦПУ, центральное процессорное устройство, микропроцессор
CPU CORE	—	ядро микропроцессора
DMA	—	Direct Memory Access
EDAC	—	Error Detection and Correction
EPS	—	Electric Power Steering
FIFO	—	First in/First out
FLASH	—	Перепрограммируемая постоянная память
FPU	—	Floating Point unit
GPIO	—	General Input-Output Port
I ² C	—	Inter-Integrated Circuit
JTAG	—	Joint Test Action Group (аппаратный интерфейс на базе стандарта IEEE 1149.1)
LDPC	—	Low Density Parity Check Code код с малой плотностью проверок на четность)
NMI	—	Non-Maskable Interrupt

Подп. и дата.	Инв. № дубл.	Взам. инв. №	Подп. и дата	AMBA AHB	—	Advanced Microcontroller Bus Architecture, Advanced High performance Bus	
				AMBA APB	—	Advanced Microcontroller Bus Architecture, Advanced Peripheral Bus	
				AHB	—	Advanced High performance Bus	
				BusMUX	—	Advanced High performance Bus	
				AHB-APB	—	Advanced High performance Bus - Advanced Peripheral Bus	
				CCSDS	—	Consultative Committee for Space Data Systems	
				CAN	—	Controller Area Network	
				CLK	—	Тактовый сигнал системной частоты	
				CPSDVSR	—	clockprescaledivisorvalue – коэффициент деления тактового сигнала	
				CPU	—	ЦПУ, центральное процессорное устройство, микропроцессор	
				CPU CORE	—	ядро микропроцессора	
				DMA	—	Direct Memory Access	
				EDAC	—	Error Detection and Correction	
				EPS	—	Electric Power Steering	
				FIFO	—	First in/First out	
				FLASH	—	Перепрограммируемая постоянная память	
				FPU	—	Floating Point unit	
				GPIO	—	General Input-Output Port	
				I ² C	—	Inter-Integrated Circuit	
JTAG	—	Joint Test Action Group (аппаратный интерфейс на базе стандарта IEEE 1149.1)					
LDPC	—	Low Density Parity Check Code код с малой плотностью проверок на четность)					
NMI	—	Non-Maskable Interrupt					
Инв. № подл.						Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист
							255
	Изм	Лист	№ докум.	Подп.	Дата		

NVIC	—	Nested Vectory Interrupt Controller
PCCU	—	Portable Command and Control Unit
QDU	—	Квадратурный преобразователь
RISC	—	Reduced instruction set computer
RST	—	сигнал сброса
SCL	—	линия тактового сигнала
SDA	—	последовательная линия данных
SOD	—	Slave mode Output Disable — запрет линии SPI_MOSI для ведомого режима
SPI	—	Serial Peripheral Interface (последовательный периферийный интерфейс)
SRAM	—	Synchronous Random Access Memory
SCR	—	SerialClockRate – скорость информационного обмена
SSE	—	Synchronous Serial Port Enable – разрешение последовательного синхронного порта
SSI	—	Synchronous Serial Interface (синхронно-последовательный интерфейс)
UART	—	Universal Asynchronous receiver-transmitter (универсальный асинхронный приёмопередатчик)
UTIME	—	Таймер временных отсчётов

Изм	Лист	№ докум.	Подп.	Дата	Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист 256						
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм		Лист		№ докум.		Подп.		Дата				
Изм												

Перечень ссылочных документов

Обозначение документов, на которые дана ссылка	Наименования документов
ГОСТ Р 52070-2003	Единая система конструкторской документации. Текстовые документы.
ГОСТ В 20.39.412 - 97	Комплексная система общих технических требований. Изделия электронной техники, квантовой электроники и электротехнические военного назначения. Общие технические требования.
АЕЯР.431290.994 ТУ	Микросхема интегральная 5023BC016 Технические условия

Ине. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.

					Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист
Изм	Лист	№ докум.	Подп.	Дата		257

Перечень изменений

Версия	Дата	Описание
2.0	22 июня 2017	Исходная версия
2.01	29 июня 2017	Внесены изменения в раздел 1.2. Исправлен рисунок 1.2.1. Исправлена таблица 1.2.1. Внесены исправления в раздел 3.1 (описание функциональных блоков). Внесены изменения в раздел 3.19. Изменён рисунок 3.19.1. Исправлены таблицы 3.19.1 – 3.19.5
2.02	07 июля 2017	Изменены рисунки: 1.2.1; 3.1.1. Исправлены таблицы: 1.2.1; 1.2.2; 1.3.3; 3.2.1; 3.3.2; 3.4.13. Внесены изменения в раздел 3.11. Изменены рисунки 3.11.1 – 3.11.13. Внесены исправления в Перечень сокращений и условных обозначений
2.03	14 июля 2017	Внесены изменения в рисунки 1.2.1 и 3.1.1
2.04	01 сентября 2017	Исправлен рисунок 1.2.1. Исправлены таблицы 1.2.1 и 1.2.2 Внесены изменения в раздел 3.1 и пункт 3.4.1.23.

Ине. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата.
Изм	Лист	№ докум.	Подп.	Дата
Микросхема интегральная 5023BC016				Лист
Техническое описание				
Версия 2.04 от 01.09.2017 г.				258

Лист регистрации изменений

[illegible]

Инв. № подл.	Подп. и дата	Взам. инв.	Инв. №	Подп. и дата.

					Микросхема интегральная 5023BC016 Техническое описание Версия 2.04 от 01.09.2017 г.	Лист
						259
Изм.	Лист	№ докум.	Подп.	Дата		