

# **МИКРОСХЕМА ИНТЕГРАЛЬНАЯ 1892ХД1Я**

**РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ**

## ОГЛАВЛЕНИЕ

1. ВВЕДЕНИЕ.....	4
1.1 Назначение .....	4
1.2 Функциональные параметры и возможности.....	5
2. СИСТЕМНАЯ ОРГАНИЗАЦИЯ МИКРОСХЕМЫ .....	6
2.1 Структурная схема.....	6
2.2 Схема синхронизации.....	8
2.3 Карта памяти .....	9
3. АДАПТЕР СОПРЯЖЕНИЯ С ШИНОЙ ВНЕШНЕГО МИКРОПРОЦЕССОРА .....	16
3.1 Назначение .....	16
3.2 Регистры МВА .....	16
3.2.1 Регистр запросов прерывания .....	16
3.2.2 Регистр маски запросов прерывания MASKR.....	18
3.2.3 Регистр буферизации адреса и данных BDR .....	18
3.2.4 Регистр BUSY .....	18
3.3 Обмен данными с микросхемой 1892ХД1Я через адаптер МВА.....	19
3.3.1 Общие положения .....	19
3.3.2 Обмен данными с использованием признака занятости BUSY .....	19
3.3.3 Обмен данными с использованием сигнала nACK .....	20
3.4 Подключение к микропроцессору нескольких микросхем 1892ХД1Я .....	23
4. КОНТРОЛЛЕР ШИНЫ РСІ .....	24
4.1 Функциональные параметры и возможности.....	24
4.2 Структурная схема.....	24
4.3 Регистры .....	25
4.3.1 Конфигурационные регистры.....	26
4.3.2 Регистры управления обменом данных .....	29
4.4 Обмен данными по шине РСІ в режиме Target (Slave) .....	34
4.5 Обмен данными с шиной РСІ в режиме Master .....	35
4.6 Обмен данными с шиной РСІ через адресное окно.....	36
4.7 Передача прерываний.....	38
4.7.1 Передача вектора прерывания из шины РСІ.....	38
4.7.2 Передача прерываний в шину РСІ.....	38
4.8 Арбитр.....	40
5. КОНТРОЛЛЕР SWIC.....	41
5.1 Структурная схема.....	41
5.2 Программная модель .....	42
5.2.1 Общие положения .....	42
5.3 Описание регистров SWIC.....	44
5.3.1 Регистр HW_VER .....	44
5.3.2 Регистр STATUS.....	44
5.3.3 Регистр RX_CODE .....	47
5.3.4 Регистр MODE_CR.....	47
5.3.5 Регистр TX_SPEED .....	49
5.3.6 Регистр TX_CODE.....	50
5.3.7 Регистр RX_SPEED .....	50
5.3.8 Регистр CNT_RX_PACK.....	51
5.3.9 Регистр CNT_RX0_PACK.....	51
5.3.10 Регистр ISR_L .....	51
5.3.11 Регистр ISR_H.....	52
5.3.12 Регистр TRUE_TIME.....	52
5.3.13 Регистр TOUT_CODE .....	52

5.3.14	Регистр ISR_tout_L.....	53
5.3.15	Регистр ISR_tout_H.....	53
5.3.16	Регистр LOG_ADDR.....	53
5.4	Логика работы SWIC.....	54
5.4.1	Установление соединения.....	54
5.4.2	Прием пакетов данных из канала связи SpaceWire.....	54
5.4.3	Передача пакетов данных в канал связи SpaceWire.....	57
5.4.4	Размещение данных и дескрипторов в памяти.....	59
5.4.5	Формат дескриптора пакета.....	60
5.4.6	Маркеры времени.....	60
5.4.7	Коды распределенных прерываний.....	61
5.4.8	Poll-коды.....	61
5.5	Логика работы DMA SWIC.....	61
5.5.1	Типы каналов.....	61
5.5.2	Регистры DMA.....	63
5.5.3	Запуск канала DMA при помощи регистра CSR.....	65
5.5.4	Выполнение процедуры самоинициализации.....	65
5.6	Примеры работы со SWIC.....	66
5.6.1	Процедура передачи пакета.....	66
5.6.2	Процедура приема пакета.....	67
6.	РЕКОМЕНДАЦИИ ПО ПРИМЕНЕНИЮ.....	68
6.1	Рекомендации по подключению failsafe резисторов.....	68
7.	ЭЛЕКТРИЧЕСКИЕ ПАРАМЕТРЫ.....	69
7.1	Электропитание.....	69
7.2	Электрические параметры при приемке и поставке.....	69
7.3	Предельно-допустимые и предельные электрические режимы эксплуатации.....	70
8.	ОПИСАНИЕ ВНЕШНИХ ВЫВОДОВ МИКРОСХЕМЫ.....	71
8.1	Назначение выводов микросхемы 1892ХД1Я.....	71
9.	ИСТОРИЯ ИЗМЕНЕНИЙ.....	81
9.1	17 июля 2013 г.....	81
9.2	22 августа 2013 г.....	81
9.3	16 сентября 2013 г.....	81
9.4	24 сентября 2013 г.....	81
9.5	01 ноября 2013 г.....	81
9.6	31 марта 2014 г.....	81
9.7	11 апреля 2014 г.....	82
9.8	23 декабря 2014 г.....	82
9.9	20 ноября 2015 г.....	82
9.10	13 ноября 2017 г.....	82
9.11	15 декабря 2017 г.....	82
9.12	1 февраля 2018 г.....	82

# 1. ВВЕДЕНИЕ

## 1.1 Назначение

Микросхема интегральная 1892ХД1Я (МСВ-01 - MultiCore Bridge) спроектирована как однокристалльный контроллер ввода-вывода на базе IP-ядерной (IP-intellectual property) платформы «МУЛЬТИКОР», разработанной в АО НПЦ «ЭЛВИС».

Микросхема 1892ХД1Я предназначена:

- для сопряжения микропроцессоров, например, сигнальных микропроцессоров семейства «Мультикор» с каналами связи SpaceWire и шиной PCI;
- для сопряжения устройств с интерфейсом PCI с каналами связи SpaceWire.

Использование микросхемы 1892ХД1Я позволяет процессору передавать и принимать данные с использованием четырех контроллеров SWIC, взаимодействовать с блоками памяти на шине PCI. Процессор обменивается данными с микросхемой 1892ХД1Я через интерфейс асинхронной памяти.

Использование микросхемы 1892ХД1Я так же возможно под управлением процессора через интерфейс PCI. Интерфейс PCI функционирует в 32-х разрядном режиме, частота 33 или 66 МГц.

Микросхема 1892ХД1Я генерирует прерывания (nINT на шине памяти и nINTA на шине PCI), причиной которых может являться установка соединения по каналам связи SpaceWire, разрыв соединения, получение маркера времени или конца пакета, завершение областей памяти, на которые настроены контроллеры SWIC и т.д..

Микросхема 1892ХД1Я обеспечивает взаимодействие вычислительных модулей в распределенных вычислительных системах, комплексах обработки сигналов и данных.

Микросхема 1892ХД1Я обеспечивает взаимодействие процессора с коммуникационной сетью на основе стандарта SpaceWire, которая, как правило, строится на основе коммутаторов-маршрутизаторов. Каналы связи SpaceWire также могут быть использованы для сопряжения с источниками/приемниками высокоскоростных потоков цифровых сигналов и данных.

Таким образом, микросхема 1892ХД1Я позволяет обеспечить взаимодействие процессоров и периферийных устройств, поддерживающих стандарты SpaceWire и PCI в широком диапазоне приложений, от передачи разнородного потока коротких пакетов с использованием технологии виртуальных каналов до непрерывного однородного потока данных.

## 1.2 Функциональные параметры и возможности

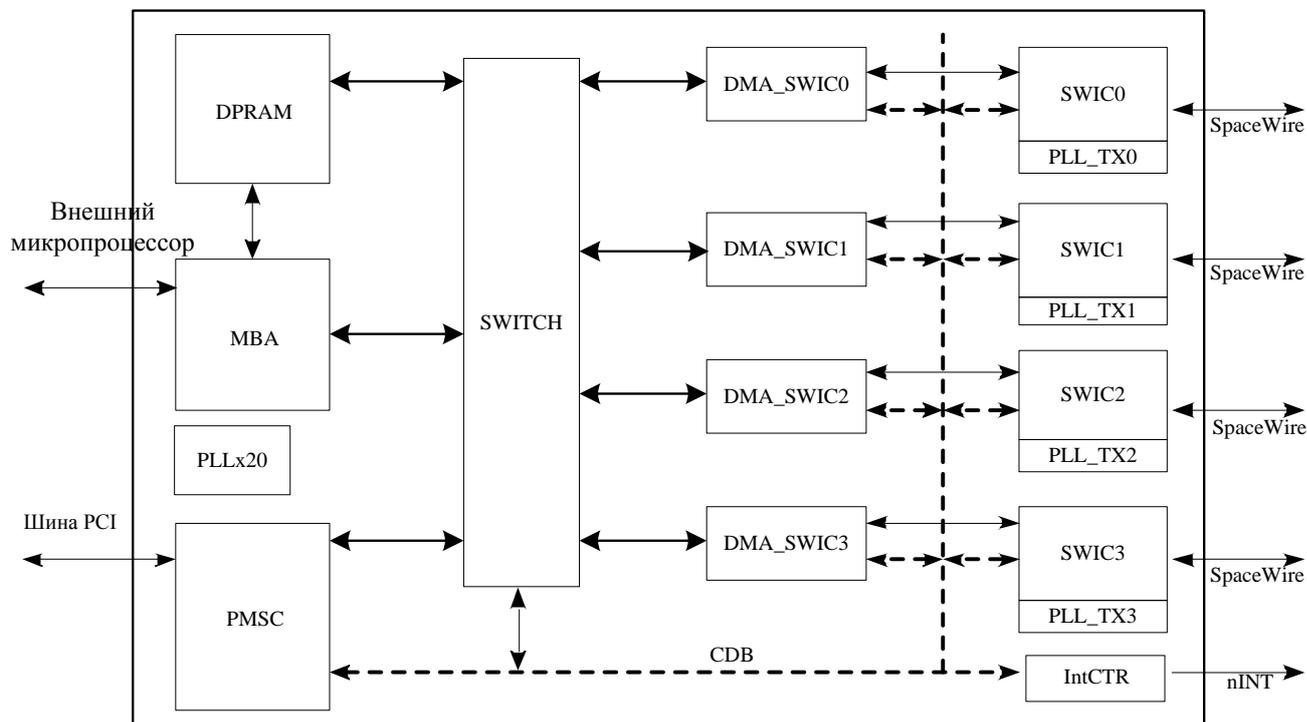
Микросхема 1892ХД1Я имеет следующие функциональные параметры и возможности:

- встроенный порт для сопряжения с внешним микропроцессором: шина данных – 32 разряда, шина адреса – 25 разрядов;
- встроенный контроллер шины PCI:
  - соответствует спецификации Local Bus Specification. Rev.2.3;
  - тактовая частота – от 33 до 66 МГц;
  - шина адреса и данных - 32 разряда;
  - обмен данными в режиме Master и Target (Slave);
  - встроенный арбитр на 5 запросов шины;
  - встроенные средства для организации мультипроцессорных систем;
  - обеспечивает обмен данными между шиной PCI и любой областью внутренней памяти и регистрами устройств;
- четыре контроллера канала связи SpaceWire:
  - соответствуют стандарту ECSS-E-50-12A:
  - скорость приема и передачи данных – от 5 до 250 Мбит/с;
  - дуплексный режим работы;
  - возможность программной настройки скорости передачи данных;
  - аппаратное детектирование ошибок связи: разъединение, ошибки чётности;
  - возможность программной адаптивной подстройки скорости;
  - обмен данными с внутренней памятью по каналам DMA 32-х разрядными словами;
  - встроенные LVDS-приемопередатчики в соответствии со стандартом ANSI/TIA/EIA-644;
  - согласующие резисторы, встроенные в LVDS приемники сопротивлением 130 +/- 10% Ом.
- встроенная память данных объемом 256 Кбайт.

## 2. СИСТЕМНАЯ ОРГАНИЗАЦИЯ МИКРОСХЕМЫ

### 2.1 Структурная схема

Структурная схема микросхемы 1892ХД1Я приведена на **Рисунок 2.1**.

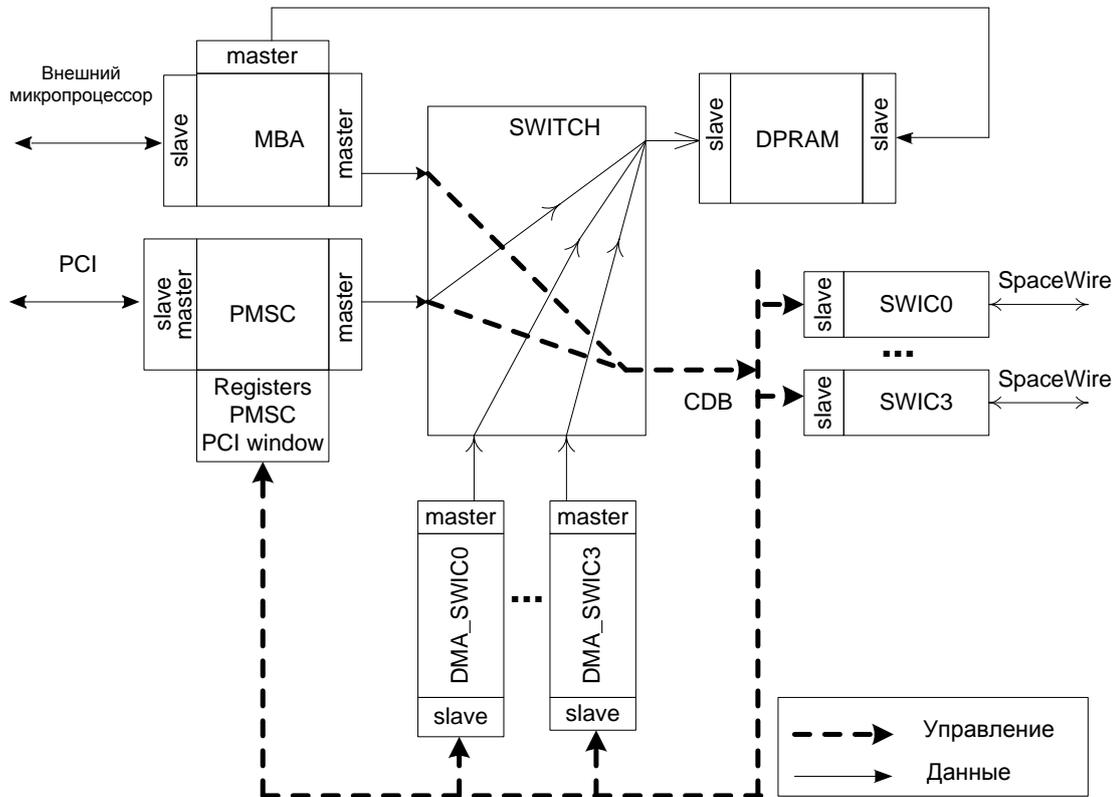


**Рисунок 2.1** Структурная схема микросхемы 1892ХД1Я

Микросхема 1892ХД1Я содержит следующие основные узлы:

- SpaceWire – канал связи SpaceWire;
- MBA - (Microprocessor Bus Adapter) - адаптер сопряжения с шинами адреса и данных внешнего микропроцессора;
- PMSC – (PCI Master/Slave Controller) - контроллер шины PCI;
- SWITCH - коммутатор данных. Обеспечивает обмен данными между DMA\_SWIC и DPRAM, между PMSC и DPRAM и доступ к шине CDB от блоков MBA и PMSC;
- SWIC – (SpaceWire Interface Controller) контроллеры SpaceWire;
- DMA\_SWIC - контроллеры DMA блоков SWIC;
- PLL\_TX – программируемый умножитель частоты. Формирует частоту для передачи данных контроллерами SWIC;
- DPRAM - двухпортовое оперативное запоминающее устройство;
- CDB – (Control Data Bus) - шина управления. Используется для обмена данными с регистрами SWIC, DMA\_SWIC, PMSC;
- IntCTR - контроллер прерываний;
- PLLx20 - программируемый умножитель частоты. Формирует системную частоту 200 МГц.

Схема потоков данных и управления внутри микросхемы 1892ХД1Я приведена на **Рисунок 2.2**.

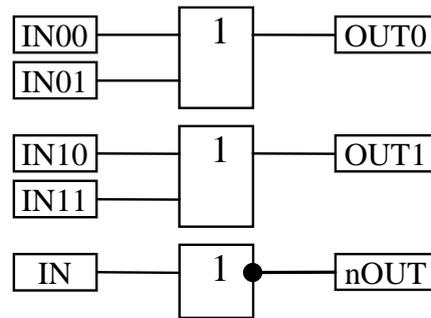


**Рисунок 2.2** Схема потоков данных и управления

В микросхеме 1892XD1Y все инициаторы обмена (master) и исполнители (slave) подключены к коммутатору SWITCH. Устройство master не может обращаться к "себе" как устройству slave. При таких обменах данные при записи теряются, а при чтении недостоверны.

Коммутатор в соответствии картой памяти соединяет устройство master, с абонентами шины CDB или с DPRAM. Абонентами шины CDB являются регистры блоков SWIC, DMA\_SWIC и PMSC. Доступ к DPRAM от внешнего микропроцессора организован через отдельный порт блока MBA.

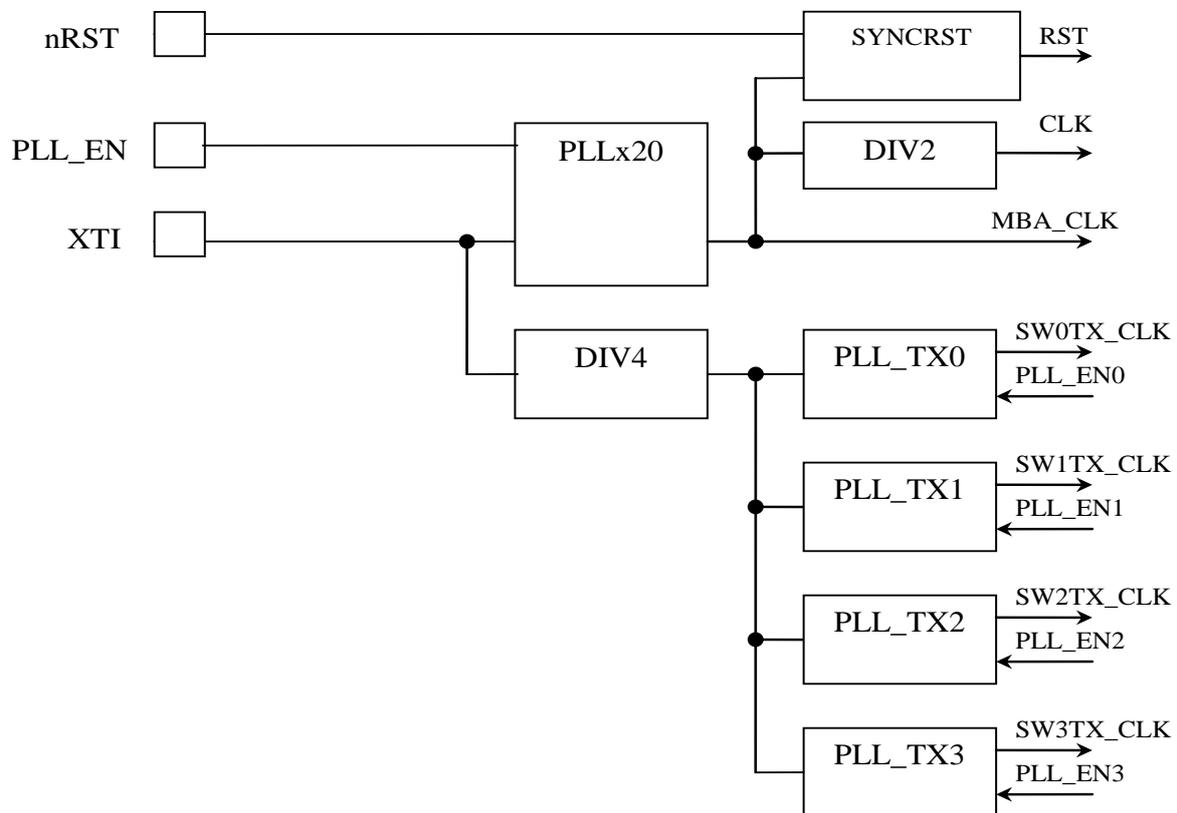
В микросхеме 1892XD1Y имеются логические элементы. Схема подключения к контактным площадкам этих элементов приведена на Рисунок 2.3. Неиспользуемые входы следует подключать к GND.



**Рисунок 2.3 Подключение логических элементов к выводам микросхемы**

## 2.2 Схема синхронизации

Схема синхронизации микросхемы 1892ХД1Я приведена на Рисунок 2.4.



**Рисунок 2.4 Схема синхронизации микросхемы 1892ХД1Я**

Синхронизация микросхемы 1892ХД1Я осуществляется от внешнего генератора, подключаемого к выводу XTI. Частота синхронизации должна быть  $10 \pm 1\%$  МГц.

Частота с вывода XTI поступает на блок PLLx20 с фиксированным коэффициентом умножения частоты равным 20, на выходе которого формируется системная частота MBA\_CLK. Разрешение умножения входной частоты блоком PLLx20 производится высоким уровнем сигнала PLL\_EN (рабочий режим). При PLL\_EN=0 (тестовый режим) выход-

ная частота PLLx20 равна входной частоте. Частота MBA\_CLK используется для тактирования MBA и DPRAM. На делителе DIV2 производится деление частоты MBA\_CLK на два и формируется частота CLK, используемая для тактирования SWITCH, SWIC, DMA SWIC и PMSC, при обменах с шиной CDB и коммутатором данных.

Со входа ХТІ частота поступает и на делитель с коэффициентом деления 4 (DIV4), после чего полученная частота 2.5 МГц поступает на входы умножителей частоты PLL\_TX3:PLL\_TX0. Управление коэффициентом умножения осуществляется программно через регистры TX\_SPEED. Каждому SWIC сопоставлен свой блок PLL\_TX. Частота, формируемая умножителем PLL\_TX, используется для синхронизации передающего тракта данных, соответствующего SWIC.

Входной сигнал установки исходного состояния nRST стробируется в блоке SYNCRST частотой MBA\_CLK. Длительность сигнала nRST – не менее 20 нс.

## 2.3 Карта памяти

Карта памяти микросхемы 1892ХД1Я при обмене данными через адаптер MBA приведена в **Таблица 2.1**.

По внешней шине A[24:0] передается адрес 32-разрядного слова. Поэтому при сопряжении микросхемы 1892ХД1Я с микропроцессорами серии «Мультикор» эту шину необходимо подключить к шине адреса микропроцессора, начиная со 2 разряда. При обмене данными через адаптер MBA, внутренний адрес микросхемы 1892ХД1Я формируется добавлением к разрядам A[22:0] двух нулей справа. В Таблица 2.1 и далее по тексту используются именно такая система обозначения адресов.

**Таблица 2.1 Карта памяти микросхемы 1892ХД1Я при обмене данными через адаптер MBA**

Диапазон адресов	Название области	Объем области, Мбайт
1FF_FFFC-1E0_0000	Резерв	2
1DF_FFFC-1C0_0000	Регистры MBA	2
1BF_FFFC-1B0_0000	Регистры DMA_SWIC3	1
1AF_FFFC-1A0_0000	Регистры SWIC3	1
19F_FFFC-190_0000	Регистры DMA_SWIC2	1
18F_FFFC-180_0000	Регистры SWIC2	1
17F_FFFC-170_0000	Регистры DMA_SWIC1	1
16F_FFFC-160_0000	Регистры SWIC1	1
15F_FFFC-150_0000	Регистры DMA_SWIC0	1
14F_FFFC-140_0000	Регистры SWIC0	1
13F_FFFC-120_0000	Регистры PMSC	2
11F_FFFC-104_0000	Резерв	,
103_FFFC-100_0000	Память DPRAM	0,256
0FF_FFFC-000_0000	Окно шины PCI	16

Карта памяти микросхемы 1892ХД1Я при обмене данными через шину PCI приведена в Таблица 2.2.

**Таблица 2.2 Карта памяти микросхемы 1892ХД1Я при обмене данными через шину PCI**

Диапазон адресов	Название области	Объем области, Мбайт
1FF_FFFC-1C0_0000	Резерв	4
01BF_FFFC-01B0_0000	Регистры DMA_SWIC3	1
01AF_FFFC-01A0_0000	Регистры SWIC3	1
019F_FFFC-0190_0000	Регистры DMA_SWIC2	1
018F_FFFC-0180_0000	Регистры SWIC2	1
017F_FFFC-0170_0000	Регистры DMA_SWIC1	1
016F_FFFC-0160_0000	Регистры SWIC1	1
015F_FFFC-0150_0000	Регистры DMA_SWIC0	1
014F_FFFC-0140_0000	Регистры SWIC0	1
013F_FFFC-0104_0000	Резерв	3,744
0103_FFFC-0100_0000	Память DPRAM	0,256
00FF_FFFC-0030_0000	Резерв	12
002F_FFFC-002F_0000	Регистры PMSC	0,064
002E_FFFC-0000_0000	Резерв	3

Перечень программно-доступных регистров микросхемы 1892ХД1Я приведен в Таблица 2.3. Адреса регистров указаны для случая обмена данными через адаптер МВА. При обмене данными с микросхемой 1892ХД1Я через шину PCI базовые адреса регистров необходимо определять по Таблица 2.2.

**Таблица 2.3 Перечень программно-доступных регистров микросхемы 1892ХД1Я**

Условное обозначение регистра	Назначение регистра	Адрес регистра
<u>Регистры МВА</u>		
BUSY	Регистр признака занятости	1C0_000C
BDR	Регистр буферных данных	1C0_0008
MASK	Регистр маски запросов прерывания	1C0_0004
QSTR	Регистр запросов прерывания	1C0_0000
<u>Регистры PMSC</u>		
Device/Vendor ID	Регистр идентификации устройства. Конфигурационный регистр шины PCI.	120_0000
Status/Command	Регистр состояния и управления. Конфигурационный регистр шины PCI.	120_0004
Class Code/Revision ID	Регистр кода. Конфигурационный регистр шины PCI.	120_0008
Latency Timer	Регистр таймера времени передачи (MLT). Конфигурационный регистр шины PCI.	120_000C
BAR	(Base Address Register) Регистр базового адреса. Конфигурационный регистр шины PCI.	120_0010
Interrupt_Line	Код прерывания. Конфигурационный регистр шины PCI.	120_003C
SEM	Регистр семафора.	120_0044
MBR_PCI	Регистр почтового ящика шины PCI	120_0048
CSR_PCI	Регистр управления и состояния шины PCI	120_004C
CSR_MASTER	Регистр управления режимом Master	120_0050
IR_MASTER	Регистр адреса памяти в режиме Master	120_0054
AR_PCI	Регистр адреса шины PCI в режиме Master	120_0058
QSTR_PCI	Регистр системных прерываний.	120_005C
MASKR_PCI	Регистр маски системных прерываний.	120_0060
STATUS_MASTER	Регистр состояния обмена с шиной PCI в режиме Master	120_0064

Условное обозначение регистра	Назначение регистра	Адрес регистра
TMR_PCI	Регистр параметров.	120_0068
CSR_WIN	Регистр управления обменом с PCI через адресное окно	120_006C
MBR_MBA	Регистр почтового ящика MBA	120_0070
<u>Регистры SWIC0</u>		
HW_VER	Регистр номера версии контроллера	140_0000
STATUS	Регистр состояния	140_0004
RX_CODE	Регистр маркера времени, принятого из канала связи SpaceWire	140_0008
MODE_CR	Регистр режима	140_000C
TX_SPEED	Регистр скорости передачи данных в канал связи SpaceWire	140_0010
TX_CODE	Регистр передаваемого маркера времени	140_0014
RX_SPEED	Регистр скорости приема данных из канала связи SpaceWire	140_0018
CNT_RX0_PACK	Регистр счетчика принятых пакетов нулевой длины (идушие подряд символы концов пакетов)	140_001C
CNT_RX_PACK	Регистр счетчика принятых пакетов ненулевой длины	140_0020
ISR_L	Регистр распределенных прерываний (младшая часть)	140_0024
ISR_H	Регистр распределенных прерываний (старшая часть)	140_0028
TRUE_TIME	Регистр достоверного маркера времени	140_002C
TOUT_CODE	Регистр размера таймаутов	140_0030
ISR_tout_L	Младшие разряды регистра флагов таймаутов ISR	140_0034
ISR_tout_H	Старшие разряды регистра флагов таймаутов ISR	140_0038
LOG_ADDR	Регистр логического адреса	140_003C
<u>Регистры SWIC1</u>		
HW_VER	Регистр номера версии контроллера	160_0000
STATUS	Регистр состояния	160_0004
RX_CODE	Регистр маркера времени, принятого из канала связи SpaceWire	160_0008
MODE_CR	Регистр режима	160_000C
TX_SPEED	Регистр скорости передачи данных в канал связи SpaceWire	160_0010
TX_CODE	Регистр передаваемого маркера времени	160_0014
RX_SPEED	Регистр скорости приема данных из канала связи SpaceWire	160_0018
CNT_RX0_PACK	Регистр счетчика принятых пакетов нулевой длины (идушие подряд символы концов пакетов)	160_001C
CNT_RX_PACK	Регистр счетчика принятых пакетов ненулевой длины	160_0020
ISR_L	Регистр распределенных прерываний (младшая часть)	160_0024
ISR_H	Регистр распределенных прерываний (старшая часть)	160_0028
TRUE_TIME	Регистр достоверного маркера времени	160_002C
TOUT_CODE	Регистр размера таймаутов	160_0030
ISR_tout_L	Младшие разряды регистра флагов таймаутов ISR	160_0034
ISR_tout_H	Старшие разряды регистра флагов таймаутов ISR	160_0038
LOG_ADDR	Регистр логического адреса	160_003C
<u>Регистры SWIC2</u>		
HW_VER	Регистр номера версии контроллера	180_0000

Условное обозначение регистра	Назначение регистра	Адрес регистра
STATUS	Регистр состояния	180_0004
RX_CODE	Регистр маркера времени, принятого из канала связи SpaceWire	180_0008
MODE_CR	Регистр режима	180_000C
TX_SPEED	Регистр скорости передачи данных в канал связи SpaceWire	180_0010
TX_CODE	Регистр передаваемого маркера времени	180_0014
RX_SPEED	Регистр скорости приема данных из канала связи SpaceWire	180_0018
CNT_RX0_PACK	Регистр счетчика принятых пакетов нулевой длины (идущие подряд символы концов пакетов)	180_001C
CNT_RX_PACK	Регистр счетчика принятых пакетов ненулевой длины	180_0020
ISR_L	Регистр распределенных прерываний (младшая часть)	180_0024
ISR_H	Регистр распределенных прерываний (старшая часть)	180_0028
TRUE_TIME	Регистр достоверного маркера времени	180_002C
TOUT_CODE	Регистр размера таймаутов	180_0030
ISR_tout_L	Младшие разряды регистра флагов таймаутов ISR	180_0034
ISR_tout_H	Старшие разряды регистра флагов таймаутов ISR	180_0038
LOG_ADDR	Регистр логического адреса	180_003C
<u>Регистры SWIC3</u>		
HW_VER	Регистр номера версии контроллера	1A0_0000
STATUS	Регистр состояния	1A0_0004
RX_CODE	Регистр маркера времени, принятого из канала связи SpaceWire	1A0_0008
MODE_CR	Регистр режима	1A0_000C
TX_SPEED	Регистр скорости передачи данных в канал связи SpaceWire	1A0_0010
TX_CODE	Регистр передаваемого маркера времени	1A0_0014
RX_SPEED	Регистр скорости приема данных из канала связи SpaceWire	1A0_0018
CNT_RX0_PACK	Регистр счетчика принятых пакетов нулевой длины (идущие подряд символы концов пакетов)	1A0_001C
CNT_RX_PACK	Регистр счетчика принятых пакетов ненулевой длины	1A0_0020
ISR_L	Регистр распределенных прерываний (младшая часть)	1A0_0024
ISR_H	Регистр распределенных прерываний (старшая часть)	1A0_0028
TRUE_TIME	Регистр достоверного маркера времени	1A0_002C
TOUT_CODE	Регистр размера таймаутов	1A0_0030
ISR_tout_L	Младшие разряды регистра флагов таймаутов ISR	1A0_0034
ISR_tout_H	Старшие разряды регистра флагов таймаутов ISR	1A0_0038

Условное обозначение регистра	Назначение регистра	Адрес регистра
LOG_ADDR	Регистр логического адреса	1A0_003C
<u>Регистры DMA SWIC0</u>		
SWIC0_RX_DESC_CSR	Регистр управления и состояния канала SWIC0_RX_DESC	150_0000
SWIC0_RX_DESC_CPR	Регистр указателя цепочки канала SWIC0_RX_DESC	150_0004
SWIC0_RX_DESC_IR	Индексный регистр канала SWIC0_RX_DESC	150_0008
SWIC0_RX_DESC_Run	Псевдорегистр управления битом RUN канала SWIC0_RX_DESC	150_000C
SWIC0_RX_DATA_CSR	Регистр управления и состояния канала SWIC0_RX_DATA	150_0040
SWIC0_RX_DATA_CPR	Регистр указателя цепочки канала SWIC0_RX_DATA	150_00444
SWIC0_RX_DATA_IR	Индексный регистр канала SWIC0_RX_DATA	150_0048
SWIC0_RX_DATA_Run	Псевдорегистр управления битом RUN канала SWIC0_RX_DATA	150_004C
SWIC0_TX_DESC_CSR	Регистр управления и состояния канала SWIC0_TX_DESC	150_0080
SWIC0_TX_DESC_CPR	Регистр указателя цепочки канала SWIC0_TX_DESC	150_0084
SWIC0_TX_DESC_IR	Индексный регистр канала SWIC0_TX_DESC	150_0088
SWIC0_TX_DESC_Run	Псевдорегистр управления битом RUN канала SWIC0_TX_DESC	150_008C
SWIC0_TX_DATA_CSR	Регистр управления и состояния канала SWIC0_TX_DATA	150_00C0
SWIC0_TX_DATA_CPR	Регистр указателя цепочки канала SWIC0_TX_DATA	150_00C4
SWIC0_TX_DATA_IR	Индексный регистр канала SWIC0_TX_DATA	150_00C8
SWIC0_TX_DATA_Run	Псевдорегистр управления битом RUN канала SWIC0_TX_DATA	150_00CC
<u>Регистры DMA SWIC1</u>		
SWIC1_RX_DESC_CSR	Регистр управления и состояния канала SWIC1_RX_DESC	170_0000
SWIC1_RX_DESC_CPR	Регистр указателя цепочки канала SWIC1_RX_DESC	170_0005
SWIC1_RX_DESC_IR	Индексный регистр канала SWIC1_RX_DESC	170_0008
SWIC1_RX_DESC_Run	Псевдорегистр управления битом RUN канала SWIC1_RX_DESC	170_000C
SWIC1_RX_DATA_CSR	Регистр управления и состояния канала SWIC1_RX_DATA	170_0040
SWIC1_RX_DATA_CPR	Регистр указателя цепочки канала SWIC1_RX_DATA	170_0044
SWIC1_RX_DATA_IR	Индексный регистр канала SWIC1_RX_DATA	170_0048
SWIC1_RX_DATA_Run	Псевдорегистр управления битом RUN канала SWIC1_RX_DATA	170_004C
SWIC1_TX_DESC_CSR	Регистр управления и состояния канала SWIC1_TX_DESC	170_0080
SWIC1_TX_DESC_CPR	Регистр указателя цепочки канала SWIC1_TX_DESC	170_0084
SWIC1_TX_DESC_IR	Индексный регистр канала SWIC1_TX_DESC	170_0088
SWIC1_TX_DESC_Run	Псевдорегистр управления битом RUN канала SWIC1_TX_DESC	170_008C
SWIC1_TX_DATA_CSR	Регистр управления и состояния канала SWIC1_TX_DATA	170_00C0

Условное обозначение регистра	Назначение регистра	Адрес регистра
SWIC1_TX_DATA_CP	Регистр указателя цепочки канала SWIC1_TX_DATA	170_00C4
SWIC1_TX_DATA_IR	Индексный регистр канала SWIC1_TX_DATA	170_00C8
SWIC1_TX_DATA_Run	Псевдорегистр управления битом RUN канала SWIC1_TX_DATA	170_00CC
<u>Регистры DMA SWIC2</u>		
SWIC2_RX_DESC_CSR	Регистр управления и состояния канала SWIC2_RX_DESC	190_0000
SWIC2_RX_DESC_CP	Регистр указателя цепочки канала SWIC2_RX_DESC	190_0004
SWIC2_RX_DESC_IR	Индексный регистр канала SWIC2_RX_DESC	190_0008
SWIC2_RX_DESC_Run	Псевдорегистр управления битом RUN канала SWIC2_RX_DESC	190_000C
SWIC2_RX_DATA_CSR	Регистр управления и состояния канала SWIC2_RX_DATA	190_0040
SWIC2_RX_DATA_CP	Регистр указателя цепочки канала SWIC2_RX_DATA	190_0044
SWIC2_RX_DATA_IR	Индексный регистр канала SWIC2_RX_DATA	190_0048
SWIC2_RX_DATA_Run	Псевдорегистр управления битом RUN канала SWIC2_RX_DATA	190_004C
SWIC2_TX_DESC_CSR	Регистр управления и состояния канала SWIC2_TX_DESC	190_0080
SWIC2_TX_DESC_CP	Регистр указателя цепочки канала SWIC2_TX_DESC	190_0084
SWIC2_TX_DESC_IR	Индексный регистр канала SWIC2_TX_DESC	190_0088
SWIC2_TX_DESC_Run	Псевдорегистр управления битом RUN канала SWIC2_TX_DESC	190_008C
SWIC2_TX_DATA_CSR	Регистр управления и состояния канала SWIC2_TX_DATA	190_00C0
SWIC2_TX_DATA_CP	Регистр указателя цепочки канала SWIC2_TX_DATA	190_00C4
SWIC2_TX_DATA_IR	Индексный регистр канала SWIC2_TX_DATA	190_00C8
SWIC2_TX_DATA_Run	Псевдорегистр управления битом RUN канала SWIC2_TX_DATA	190_00CC
<u>Регистры DMA SWIC3</u>		
SWIC3_RX_DESC_CSR	Регистр управления и состояния канала SWIC3_RX_DESC	1B0_0000
SWIC3_RX_DESC_CP	Регистр указателя цепочки канала SWIC3_RX_DESC	1B0_0004
SWIC3_RX_DESC_IR	Индексный регистр канала SWIC3_RX_DESC	1B0_0008
SWIC3_RX_DESC_Run	Псевдорегистр управления битом RUN канала SWIC3_RX_DESC	1B0_000C
SWIC3_RX_DATA_CSR	Регистр управления и состояния канала SWIC3_RX_DATA	1B0_0040
SWIC3_RX_DATA_CP	Регистр указателя цепочки канала SWIC3_RX_DATA	1B0_0044
SWIC3_RX_DATA_IR	Индексный регистр канала SWIC3_RX_DATA	1B0_0048
SWIC3_RX_DATA_Run	Псевдорегистр управления битом RUN канала SWIC3_RX_DATA	1B0_004C
SWIC3_TX_DESC_CSR	Регистр управления и состояния канала SWIC3_TX_DESC	1B0_0080
SWIC3_TX_DESC_CP	Регистр указателя цепочки канала SWIC3_TX_DESC	1B0_0084
SWIC3_TX_DESC_IR	Индексный регистр канала SWIC3_TX_DESC	1B0_0088

Условное обозначение регистра	Назначение регистра	Адрес регистра
SWIC3_TX_DESC_Run	Псевдорегистр управления битом RUN канала SWIC3_TX_DESC	1B0_008C
SWIC3_TX_DATA_CSR	Регистр управления и состояния канала SWIC3_TX_DATA	1B0_00C0
SWIC3_TX_DATA_CRP	Регистр указателя цепочки канала SWIC3_TX_DATA	1B0_00C4
SWIC3_TX_DATA_IR	Индексный регистр канала SWIC3_TX_DATA	1B0_00C8
SWIC3_TX_DATA_Run	Псевдорегистр управления битом RUN канала SWIC3_TX_DATA	1B0_00CC

## 3. АДАПТЕР СОПРЯЖЕНИЯ С ШИНОЙ ВНЕШНЕГО МИКРОПРОЦЕССОРА

### 3.1 Назначение

Адаптер сопряжения с шиной данных внешнего микропроцессора (МВА) предназначен для обмена данными с регистрами SWIC, PMSC, памятью DPRAM, а также с шиной PCI в режиме Master через адресное окно. Используя доступ к регистрам PMSC, через МВА можно выполнять передачу блока данных по шине PCI в режиме Master (см. 4.5).

Обмен данными через МВА выполняется 32-разрядными словами. При записи данных в память DPRAM при помощи сигналов  $nWE[3:0]$  можно записать от 1 до 4 байт 32-разрядного слова.

Следует иметь ввиду, что по шине  $A[24:0]$  передается адрес 32-разрядного слова. Поэтому при сопряжении микросхемы 1892ХД1Я с микропроцессорами серии «Мультикор» эту шину необходимо подключить к шине адреса микропроцессора, начиная со 2 разряда.

### 3.2 Регистры МВА

#### 3.2.1 Регистр запросов прерывания

Формат регистра QSTR приведен в Таблица 3.1.

**Таблица 3.1 Назначение разрядов регистра QSTR**

Номер разряда	Условное обозначение	Назначение
Прерывания от PMSC		
31	MASTER_DONE	Окончание передачи блока данных по шине PCI в режиме Master
30	MASTER_ERROR	Обнаружение ошибки при передаче блока данных по шине PCI в режиме Master
29	MASTER_WMARK	Предварительное уведомление о передаче по шине PCI в режиме Master заданного полем WaterMark регистра TMR_PCI количества слов
28	INT_MBR	Признак записи данных в регистр почтового ящика MBR_PCI контроллера PMSC со стороны шины PCI. Данный бит обнуляется при чтении содержимого регистра MBR_PCI
Прерывания от DMA SWIC3		
27	SWIC3_TX_DATA	Окончание передачи блока данных или цепочки блоков данных каналом DMA SWIC3_TX_DATA
26	SWIC3_TX_DESC	Окончание передачи блока данных или цепочки блоков данных каналом DMA SWIC3_TX_DESC
25	SWIC3_RX_DATA	Окончание передачи блока данных или цепочки блоков данных каналом DMA SWIC3_RX_DATA
24	SWIC3_RX_DESC	Окончание передачи блока данных или цепочки блоков данных каналом DMA SWIC3_RX_DESC
Прерывания от DMA SWIC2		
23	SWIC2_TX_DATA	Окончание передачи блока данных или цепочки блоков данных каналом DMA SWIC2_TX_DATA

Номер разряда	Условное обозначение	Назначение
22	SWIC2_TX_DESC	Окончание передачи блока данных или цепочки блоков данных каналом DMA SWIC2_TX_DESC
21	SWIC2_RX_DATA	Окончание передачи блока данных или цепочки блоков данных каналом DMA SWIC2_RX_DATA
20	SWIC2_RX_DESC	Окончание передачи блока данных или цепочки блоков данных каналом DMA SWIC2_RX_DESC
Прерывания от DMA SWIC1		
19	SWIC1_TX_DATA	Окончание передачи блока данных или цепочки блоков данных каналом DMA SWIC1_TX_DATA
18	SWIC1_TX_DESC	Окончание передачи блока данных или цепочки блоков данных каналом DMA SWIC1_TX_DESC
17	SWIC1_RX_DATA	Окончание передачи блока данных или цепочки блоков данных каналом DMA SWIC1_RX_DATA
16	SWIC1_RX_DESC	Окончание передачи блока данных или цепочки блоков данных каналом DMA SWIC1_RX_DESC
Прерывания от DMA SWIC0		
15	SWIC0_TX_DATA	Окончание передачи блока данных или цепочки блоков данных каналом DMA SWIC0_TX_DATA
14	SWIC0_TX_DESC	Окончание передачи блока данных или цепочки блоков данных каналом DMA SWIC0_TX_DESC
13	SWIC0_RX_DATA	Окончание передачи блока данных или цепочки блоков данных каналом DMA SWIC0_RX_DATA
12	SWIC0_RX_DESC	Окончание передачи блока данных или цепочки блоков данных каналом DMA SWIC0_RX_DESC
Прерывания от SWIC3		
11	SWIC3_TIME	Получен управляющий код
10	SWIC3_ERR	Ошибка в канале приема
9	SWIC3_LINK	Установлено соединение/получен пакет
Прерывания от SWIC2		
8	SWIC2_TIME	Получен управляющий код
7	SWIC2_ERR	Ошибка в канале приема
6	SWIC2_LINK	Установлено соединение/получен пакет
Прерывания от SWIC1		
5	SWIC1_TIME	Получен управляющий код
4	SWIC1_ERR	Ошибка в канале приема
3	SWIC1_LINK	Установлено соединение/получен пакет
Прерывания от SWIC0		
2	SWIC0_TIME	Получен управляющий код
1	SWIC0_ERR	Ошибка в канале приема
0	SWIC0_LINK	Установлено соединение/получен пакет

Все сигналы внутренних прерываний поступают на входы доступного только по чтению регистра QSTR. Исходное состояние QSTR – все нули (нет запросов прерывания). Все незамаскированные прерывания объединяются по «или» и поступают на внешний вывод nINT.

В процессе обслуживания прерывания необходимо проанализировать состояние устройства для определения причины его возникновения. Сброс прерывания осуществляется в момент исключения причины возникновения данного прерывания. Например, прерывание от SWIC сбрасывается при записи единиц в соответствующие разряды регистра SWIC\_STATUS.

Обмен данными с регистрами осуществляется 32-разрядными словами. Если разряды регистра не используются, то из них считываются нули. При записи в этих разрядах необходимо указывать нули.

### 3.2.2 Регистр маски запросов прерывания MASKR

Каждое внутреннее прерывание микросхемы 1892XD1Я маскируется при помощи доступного по записи и чтению регистра маски MASKR, формат которого аналогичен формату регистра QSTR. В исходном состоянии MASKR=0 (все внутренние прерывания замаскированы).

### 3.2.3 Регистр буферизации адреса и данных BDR

32-разрядный регистр BDR предназначен для выполнения процедуры чтения данных из регистров SWIC, PMSC и адресного окна шины PCI.

Регистр BDR доступен по записи и чтению со стороны шины обмена данными с внешним микропроцессором. Разряды регистра BDR соответствуют разрядам шины данных D[31:0].

Исходное содержимое регистра BDR – все нули.

### 3.2.4 Регистр BUSY

Регистр BUSY предназначен для синхронизации обмена данными с регистрами SWIC, PMSC, а также с шиной PCI в режиме Master через адресное окно.

Разряд 31 этого регистра определяет полярность сигнала nACK.

Формат регистра BUSY приведен в Таблица 3.2.

**Таблица 3.2 Формат регистра BUSY**

Номер разряда	Условное обозначение	Назначение	Доступ
31	ACK	Полярность сигнала nACK: 1 – высокий активный уровень; 0 – низкий активный уровень	RW
30:1	-	Не используется	R
0	BUSY	Признак занятости MBA при обмене данными с регистрами SWIC, PMSC или с шиной PCI в режиме Master: 0 – MBA не занят; 1 – MBA занят. Устанавливается в 1: при инициализации процедуры записи данных в регистры SWIC, PMSC или адресное окно шины PCI; при записи адреса в регистр BDR для инициализации процедуры чтения данных из регистров SWIC, PMSC или адресного окна шины PCI. Устанавливается в 0: при завершении процедуры записи данных в регистры SWIC, PMSC или адресное окно шины PCI; при завершении процедуры чтения данных из регистров SWIC, PMSC, адресного окна шины PCI и записи этих данных в регистр BDR	R

Регистр BUSY доступен по чтению и по записи разряда 31. Исходное состояние – 0.

### 3.3 Обмен данными с микросхемой 1892ХД1Я через адаптер МВА

#### 3.3.1 Общие положения

Через адаптер МВА доступны регистры SWIC, PMSC, память DPRAM, а также шина PCI в режиме Master через адресное окно.

Обмен данными с регистрами SWIC, PMSC можно выполнять двумя способами:

- с использованием признака занятости BUSY;
- с использованием сигнала nACK.

#### 3.3.2 Обмен данными с использованием признака занятости BUSY

Обмен данными с регистрами МВА и памятью DPRAM выполняется за одну операцию записи или чтения.

Запись данных в регистры SWIC, PMSC и адресное окно шины PCI выполняется за 2 операции:

- операция чтения: опрос BUSY (ожидание BUSY=0);
- операция записи данных. Адрес операции определяется в соответствии с Таблица 2.1.

Чтение данных из регистров SWIC, PMSC и адресного окна шины PCI выполняется за 4 операции:

- операция чтения: опрос BUSY=0 (ожидание BUSY=0);
- операция записи адреса регистра (или окна шины PCI) в регистр BDR. После выполнения этой операции инициируется процедура чтения содержимого регистра (окна шины PCI). Результат выполнения процедуры записывается в регистр BDR;
- операция чтения: опрос BUSY=0 (ожидание BUSY=0);
- операция чтения содержимого регистра BDR.

Временная диаграмма операции записи данных в микросхему 1892ХД1Я через МВА приведена на Рисунок 3.1.

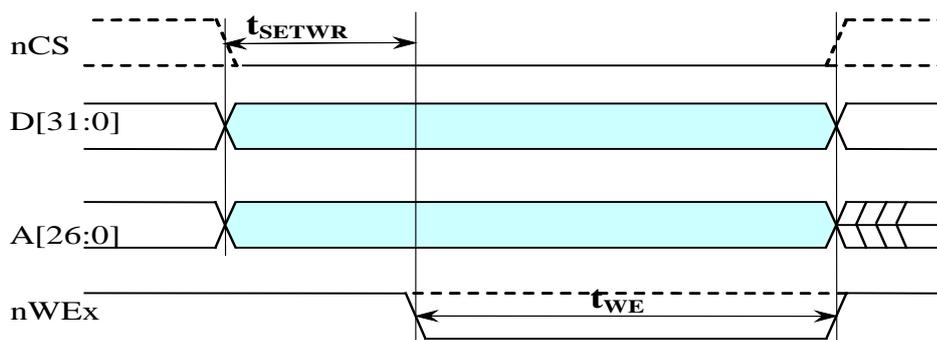


Рисунок 3.1. Временная диаграмма записи данных в микросхему 1892ХД1Я

Временная диаграмма операции чтения данных из микросхемы 1892ХД1Я через МВА приведена на Рисунок 3.2.

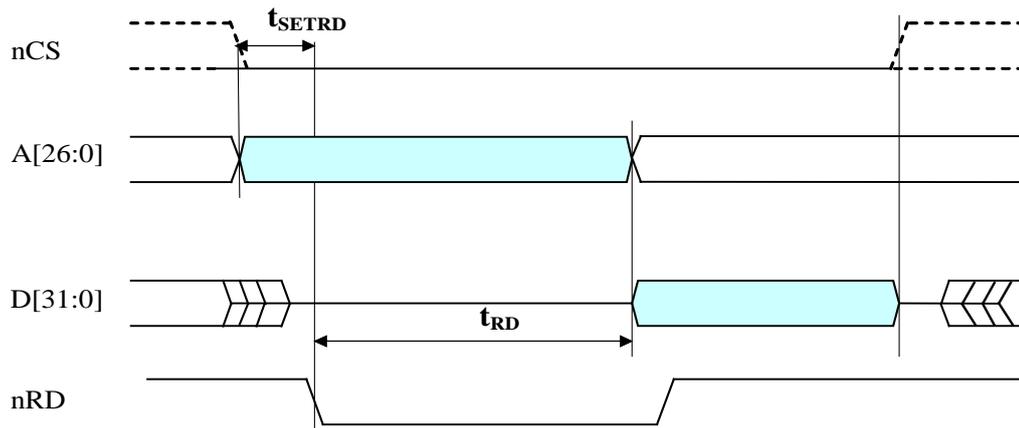


Рисунок 3.2. Временная диаграмма чтения данных из микросхемы 1892ХД1Я  
Временные параметры операций записи и чтения приведены в Таблица 3.3.

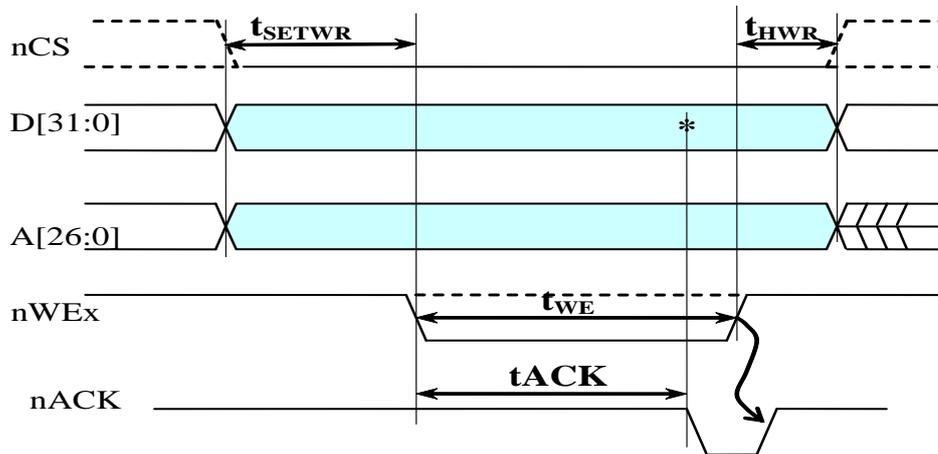
Таблица 3.3. Временные параметры операций записи и чтения

Наименование параметра, единица измерения	Буквенное обозначение	Норма	
		не менее	не более
Время установки данных, адреса и сигнала выборки до сигнала разрешения записи, нс	$t_{SETWR}$	5	-
Время установки адреса и сигнала выборки до сигнала разрешения чтения данных, нс	$t_{SETRD}$	0	-
Длительность сигнала записи данных, нс	$t_{WE}$	10,0	-
Задержка чтения данных, нс	$t_{RD}$	-	14

### 3.3.3 Обмен данными с использованием сигнала nACK

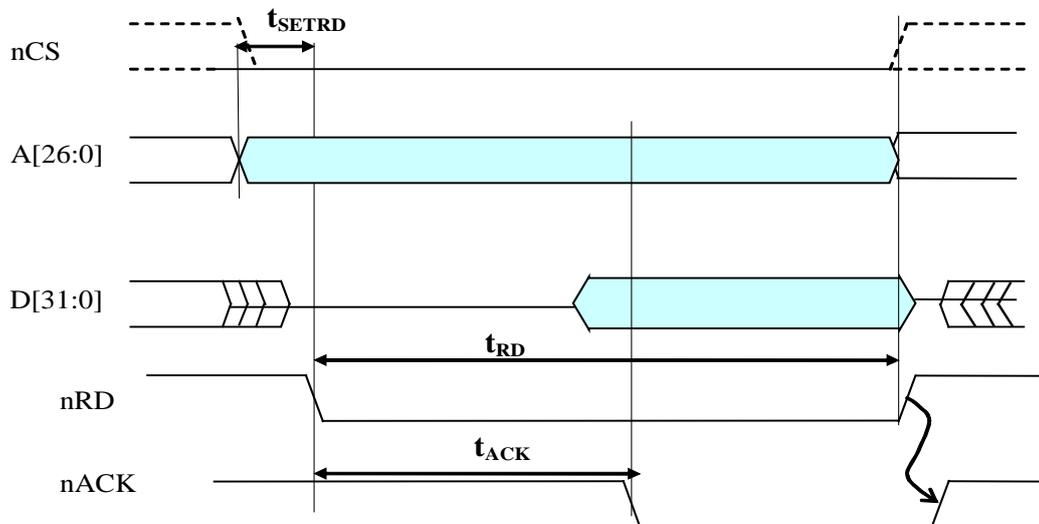
В этом случае все обмены данными с МВА выполняется за одну операцию записи или чтения.

Временная диаграмма операции записи данных в микросхему 1892ХД1Я через МВА приведена на Рисунок 3.3.



**Рисунок 3.3** Временная диаграмма записи данных в микросхему 1892XD1A

Временная диаграмма операции чтения данных из микросхемы 1892XD1A через MBA приведена на Рисунок 3.4.



**Рисунок 3.4** Временная диаграмма чтения данных из микросхемы 1892XD1A

Временные параметры операций записи и чтения приведены в Таблица 3.4.

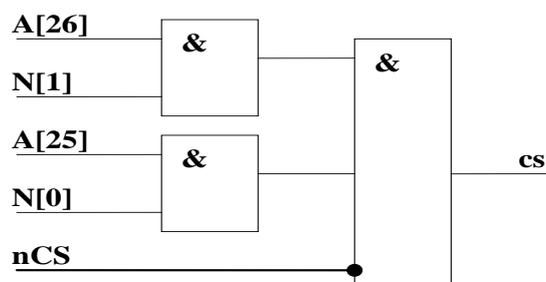
**Таблица 3.4** Временные параметры операций записи и чтения

Наименование параметра, единица измерения	Буквенное обозначение	Норма	
		не менее нс	не более нс
Время установки данных, адреса и сигнала выборки до сигнала разрешения записи	$t_{SETWR}$	5	-
Время установки адреса и сигнала выборки до сигнала разрешения чтения данных	$t_{SETRD}$	0	-
Длительность сигнала разрешения записи данных	$t_{WE}$	20,0	-

Наименование параметра, единица измерения	Буквенное обозначение	Норма	
		не менее нс	не более нс
Длительность сигнала разрешения чтения данных	$t_{RD}$	25	-
Задержка чтения данных	$t_{ACK}$	-	20
Время удержания данных, адреса и сигнала выборки после снятия сигнала разрешения записи	$t_{HWR}$	5	-

### 3.4 Подключение к микропроцессору нескольких микросхем 1892ХД1Я

Микросхема 1892ХД1Я имеет входы  $N[1:0]$  для задания номера микросхемы. Поэтому имеется возможность подключать к микропроцессору до четырех микросхем 1892ХД1Я. При обращении к микросхеме два старших бита адреса  $A[26:25]$  необходимо устанавливать соответственно подключению входов  $N[1:0]$ . На Рисунок 3.5 приведена схема сравнения адреса.



**Рисунок 3.5 Дешифрация номера микросхемы**

При соединении двух и более микросхем необходимо присваивать им различающиеся номера, например  $N[00]$ ,  $N[01]$ .

## 4. КОНТРОЛЛЕР ШИНЫ PCI

### 4.1 Функциональные параметры и возможности

Контроллер шины PCI (PMSC – PCI Master-Slave controller) имеет следующие функциональные параметры и возможности:

- соответствует спецификации Local Bus Specification. Rev. 2.3;
- тактовая частота – от 33 до 66 МГц;
- шина адреса и данных - 32 разряда;
- обмен данными в режиме Master и Target (Slave);
- встроенный арбитр на 5 запросов шины;
- встроенные средства для организации мультипроцессорных систем;
- обеспечивает обмен данными между шиной PCI и любой областью внутренней памяти и регистрами устройств.

### 4.2 Структурная схема

Структурная схема PMSC приведена на Рисунок 4.1.

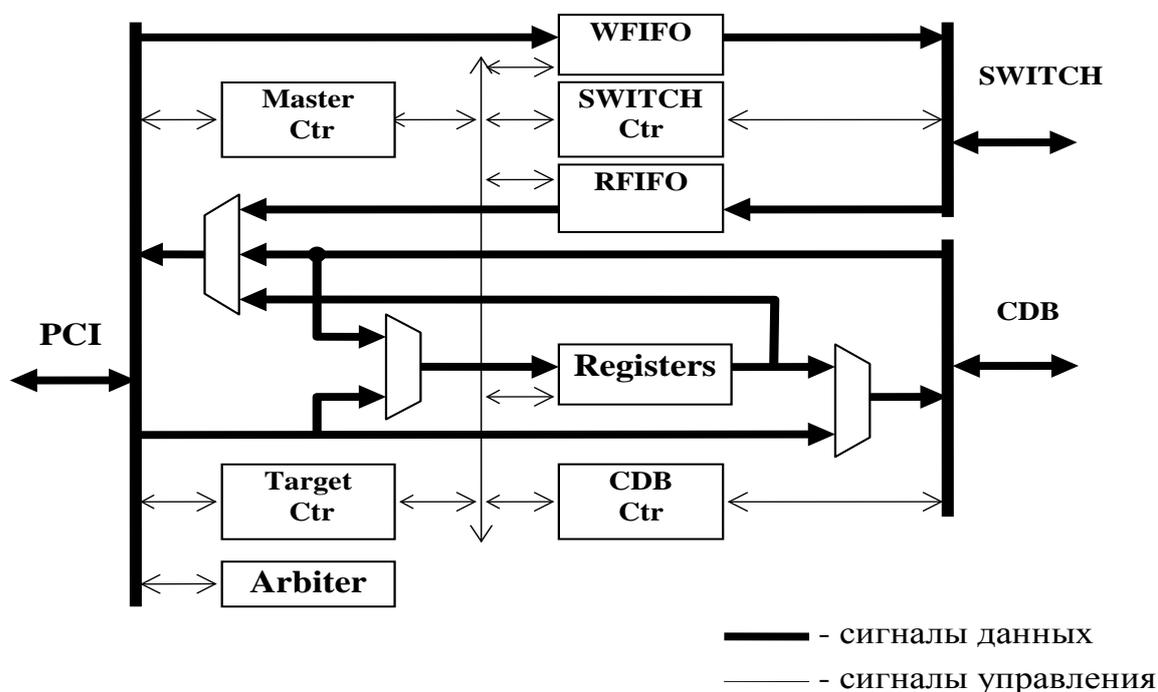


Рисунок 4.1. Структурная схема PMSC

В состав PMSC входят следующие основные узлы и компоненты:

- Registers - блок регистров, включающий:
  - конфигурационные регистры шины PCI: Device ID/Vendor ID, Status/Command, Class Code/Revision ID, Subsystem ID/Subsystem Vendor ID, BAR, Latency Timer, Interrupt Line;

- регистры управления обменом: AR\_PCI, IR\_MASTER, CSR\_MASTER, CSR\_PCI, STATUS\_MASTER, TMR\_PCI, CSR\_WIN;
- регистры передачи векторов прерывания: MBR\_MBA, MBR\_PCI и SEM;
- регистры управления прерываниями: регистр системных прерываний QSTR\_PCI и регистр маскирования прерываний MASKR\_PCI.
- CDB Ctr - блок управления обменом данными с регистрами по шине CDB;
- SWITCH Control - блок управления обменом данными с коммутатором SWITCH;
- Target Ctr – блок управления передачи данных между шиной PCI и абонентами микросхемы 1892XD1Я через коммутатор SWITCH в режиме Target (Slave);
- Master Ctr – блок управления передачи данных между шиной PCI и абонентами микросхемы 1892XD1Я через коммутатор SWITCH в режиме Master;
- WFIFO – буфер FIFO объемом 16 32-разрядных слов для передачи данных из шины PCI в коммутатор SWITCH;
- RFIFO – буфер FIFO объемом 16 32-разрядных слов для передачи данных из коммутатора SWITCH в шину PCI;
- Arbiter – арбитр шины PCI.

### 4.3 Регистры

Перечень регистров PMSC, доступных со стороны шин PCI и CDB, приведен в Таблица 4.1.

**Таблица 4.1 Перечень программно доступных регистров PMSC**

Условное обозначение регистра	Название регистра	Смещение адреса	Исходное состояние
Device ID/ Vendor ID	Регистр идентификации устройства	0x00	0x680C2001
Status/Command	Регистр состояния и управления	0x04	0x02800000
Class Code/Revision ID	Регистр кода классификации	0x08	0x07800001
Latency Timer	Регистр времени транзакции в режиме Master	0x0C	0x00000000
BAR (Base Address Register )	Регистр базового адреса	0x10	0x00000008
Subsystem ID/ Subsystem Vendor ID	Регистр идентификации подсистемы	0x2C	0x00000002
Interrupt_Line	Код прерывания	0x3C	0x01200100
SEM	Регистр семафора	0x44	0x00000000
MBR_PCI	Регистр почтового ящика шины PCI	0x48	0x00000000
CSR_PCI	Регистр управления и состояния шины PCI	0x4C	0x00000000
CSR_MASTER	Регистр управления режимом Master	0x50	0x00000000
IR_MASTER	Регистр адреса памяти в режиме Master	0x54	0x00000000
AR_PCI	Регистр адреса шины PCI в режиме Master	0x58	0x00000000
QSTR_PCI	Регистр системных прерываний.	0x5C	0x00000000
MASKR_PCI	Регистр маскирования прерываний	0x60	0x00000000
STATUS_MASTER	Регистр состояния обмена с шиной PCI в режиме Master	0x64	0x00000000
TMR_PCI	Регистр параметров	0x68	0x00000000

Условное обозначение регистра	Название регистра	Смещение адреса	Исходное состояние
CSR_WIN	Регистр управления обменом с PCI через адресное окно	0x6C	0x00000000
MBR_MBA	Регистр почтового ящика MBA	0x70	0x00000000

Смещение адреса определяется разрядами адреса 7:0 шин CDB и PCI и указано относительно базового адреса PMSC в адресном пространстве микросхемы 1892XD1Я (см. Таблица 2.1 и Таблица 2.2).

Обмен данными с регистрами осуществляется 32-разрядными словами. Если разряды регистра не используются, то из них считываются нули. При записи в этих разрядах необходимо указывать нули.

По шине CDB все регистры доступны по записи и чтению, кроме регистров QSTR\_PCI и STATUS\_MASTER. Эти регистры доступны только для чтения.

По шине PCI доступ к регистрам PMSC осуществляется в режиме Target по командам Configuration Read, Configuration Write в области адресов Type 0 и по командам Memory Read, Memory Write. При обмене данными с регистрами PMSC транзакция завершается сигнализацией Disconnect при передаче первого слова.

По чтению все регистры доступны по командам Configuration Read, Memory Read.

Регистры Status/Command, BAR, Interrupt\_Line, IR\_MASTER, AR\_PCI, MASKR\_PCI, MBR\_MBA доступны по записи по командам Configuration Write, Memory Write.

Регистры CSR\_MASTER, CSR\_PCI, MBR\_PCI, SEM, TMR\_PCI доступны для записи по команде Memory Write.

При описании полей и значений регистров используются обозначения:

- R – разрешено только чтение;
- RW – разрешены чтение и запись;
- RW0 – разрешены чтение и запись, при записи единицы разряд обнуляется;
- [i] – номер разряда;
- i:j – неразрывная группа разрядов, i – старший разряд группы, j – младший;
- 0x – далее следует код в шестнадцатеричной системе счисления;
- PCLK – тактовая частота шины PCI;
- AD – разряды адреса/данных шины PCI.

### 4.3.1 Конфигурационные регистры

#### 4.3.1.1 Регистры Device/Vendor ID, Class Code/Revision ID и Subsystem ID/Subsystem Vendor ID

32-разрядные регистры Device/Vendor ID, Class Code/Revision ID и Subsystem ID/Subsystem Vendor ID предназначены для хранения кодов в соответствии со спецификацией PCI. Исходное состояние регистров: Device/Vendor ID – 0x680c2001, Class Code/Revision ID – 0x07800001, Subsystem ID/Subsystem Vendor ID – 0x00000002.

По шине PCI эти регистры доступны только по чтению, их состояние можно изменить только через MBA при инициализации PMSC.

### 4.3.1.2 Регистр BAR

Регистр BAR определяет базовый адрес PMSC на шине PCI в режиме Target при выполнении команд Memory Read, Memory Write.

Разряды 31-26 регистра BAR доступны по записи и чтению. Разряды 25-0 этого регистра доступны только по чтению кода 000\_0008, что является индикатором пространства памяти объемом 64 Мбайт. Содержимое этого регистра определяется программно при инициализации PMSC через MBA или внешним контроллером PCI.

PMSC выполняет команды Memory Read, Memory Write при  $AD[31:26]=BAR[31:26]$  в фазе адреса шины PCI. При невыполнении этого условия PMSC инициирует завершение команд Memory Read, Memory Write по условию “Master-abort” установкой высокого уровня сигнала nDEVSEL.

### 4.3.1.3 Регистр Status/Command

Формат регистра Status/Command приведен в Таблица 4.2.

**Таблица 4.2 Формат регистра Status/Command**

Номер разряда	Условное обозначение	Описание	Доступ
31	Detected Parity Error	Признак обнаружения ошибки чётности. В режиме Target устанавливается при обнаружении ошибки чётности либо в фазе адреса, либо в фазе данных при выполнении транзакции запись. В режиме Master устанавливается при обнаружении ошибки чётности в фазе данных при выполнении транзакции чтение	RW0
30	-	Не используется.	
29	Received Master Abort	Транзакция в режиме Master завершается установкой признака Received Master Abort, если в течение пяти тактов частоты PCLK после выдачи nFRAME = 0 сохранялся высокий уровень сигнала nDEVSEL	RW0
28	Received Target Abort	Транзакция в режиме Master завершается установкой признака Received Target Abort при обнаружении в фазе передачи данных ситуации nSTOP = 0, nDEVSEL = 1	RW0
27	Signaled Target Abort	Признак Signaled Target Abort устанавливается в режиме Target, если в данном регистре бит Parity Error Response установлен в 1, в регистре CSR_PCI бит Target Parity Stop установлен в 1 и обнаружена ошибка чётности в фазе адреса. При этом PMSC завершает транзакцию установкой nSTOP = 0, nDEVSEL = 1	RW0
26:25	DEVSEL timing	Конфигурационный параметр. Определяет задержку выдачи сигнала nDEVSEL в тактах PCLK. PMSC обеспечивает задержку класса “medium”: сигнал nDEVSEL выдается через один такт после обнаружения низкого уровня сигнала nFRAME	R
24	Master Data Parity Error	Признак выдачи или обнаружения сигнала nPERR в режиме Master при условии Parity Error Response = 1	RW0

Номер разряда	Условное обозначение	Описание	Доступ
23	Fast Back-to-Back Capable	Конфигурационный параметр. Определяет способность выполнять транзакции типа “Fast Back-to-Back” если они выполняются не одним и тем же устройством	R
22:20	-	Не используется	R
19	Interrupt Status	Этот бит отражает наличие незамаскированных прерываний в PMSC. Только тогда когда в данном регистре бит Interrupt Disable установлен в 0, а бит Interrupt Status установлен в 1 будет установлен низкий уровень сигнала nINTA . Установка бита Interrupt Disable в 1 не влияет на состояние бита Interrupt Status	R
18:11	-	Не используется	R
10	Interrupt Disable	Запрещение формирования сигнала nINTA: при Interrupt Disable = 1 на выводе nINTA устанавливается пассивный высокий уровень; при Interrupt Disable = 0 и Interrupt Status =1 на выводе nINTA устанавливается активный низкий уровень.	RW
9:7	-	Не используется	R
6	Parity Error Response	Разрешение формирование сигнала nPERR	RW
5:3	-	Не используются	R
2	Bus Master	Разрешение работы на шине PCI в режиме Master	RW
1	Memory Space	Разрешение выполнение в режиме Target команд Memory Read, Memory Write	RW
0	-	Не используется	R

Разряды [29:28] обнуляются при запуске на шине PCI транзакции в режиме Master.

#### 4.3.1.4 Регистр Latency Timer

Формат регистра Latency Timer приведен в Таблица 4.3

**Таблица 4.3 Формат регистра Latency Timer**

Номер разряда	Условное обозначение	Назначение	Доступ
31:16	-	Не используется	R
15:8	MLT	Конфигурационная переменная. Определяет время в тактах PCLK, отведенное PMSC для выполнения транзакции в режиме Master. Устанавливается при инициализации шины PCI	RW
7:0	-	Не используется	R

#### 4.3.1.5 Регистр Interrupt Line

Формат регистра Interrupt Line приведен в Таблица 4.4

**Таблица 4.4 Формат регистра Interrupt Line**

Номер разряда	Условное обозначение	Назначение	Доступ
31:24	Max_Lat	Определяет максимальное время между двумя передачами данных PMSC по шине PCI. Max_Lat = 0x01. Цена одного разряда - 0,25 мкс	R

Номер разряда	Условное обозначение	Назначение	Доступ
23:16	Min_Gnt	Содержит минимальную величину времени, на которую PMSC занимает шину PCI при передаче данных. Min_Gnt = 0x20. Цена одного разряда - 0,25 мкс	R
15:8	Interrupt Pin	Указывает, что выход прерывания PMSC подключен к линии nINTA. Interrupt Pin = 0x01	R
7:0	Interrupt Line	Используется для реализации системных функций на PCI. Устанавливается при инициализации шины PCI	RW

## 4.3.2 Регистры управления обменом данными

### 4.3.2.1 Регистр CSR\_MASTER

Регистр CSR\_MASTER предназначен для задания параметров и режимов передачи блока данных по шине PCI в режиме Master.

Запуск процедуры передачи блока данных осуществляется записью 1 в разряд 0 регистра CSR\_MASTER. Запись в регистр CSR\_MASTER разрешена при CSR\_MASTER [0]=0.

Формат регистра CSR\_MASTER приведен в Таблица 4.5.

**Таблица 4.5 Формат регистра CSR\_MASTER**

Номер разряда	Условное обозначение	Назначение
31:16	WC	Размер блока данных – количество 32-разрядных слов, которое должен передать PMSC в режиме Master: 0x0 – 1 слово, 0x1 – 2 слова, ..., 0xFFFF - 65536 слов. Содержимое этого поля в процессе передачи данных не изменяется
15	DONE	Индикатор завершения процедуры передачи блока данных. Устанавливается в 0 при записи 1 в бит RUN
14	Fatal Error	Индикатор останова передачи блока данных по фатальной ошибке. Устанавливается в 0 при записи 1 в бит RUN
13	Break Done	Индикатор выполнения программного останова передачи блока данных Устанавливается в 0 при записи 1 в бит RUN
12	WINDOW	Индикатор выполнения обмена через адресное окно. Устанавливается в 1 при выполнении обмена через адресное окно. Устанавливается в 0 при записи 1 в бит RUN
11:8	WNM	Количество слов, которое должно накопиться в буфере WFIFO для передачи очередной порции данных в коммутатор SWITCH в режиме Master. Определяется из системных соображений, с точки зрения максимальной скорости передачи данных. Допустимые значения: 0, 7, 0xF. Если WC меньше или равно 0xF, необходимо определить WNM=0xF, в противном случае – выбрать WNM=7 или WNM=0. При WNM = 0,1, 2, 3 передача в коммутатор SWITCH производится пачками по 4 слова
7	sel_MILat	Разрешение изменения параметра "Master Initial Latency".
6	sel_MSLat	Разрешение изменения параметра "Master Subsequent Latency"
5	sel_IRDLat	Разрешение изменения параметра "nIRDY Latency"

Номер разряда	Условное обозначение	Назначение
4:1	CMD	<p>Тип команды при передаче данных в режиме Master:</p> <p>0010 – I/O Read;            0011 – I/O Write;            0110 – Memory Read;            0111 – Memory Write;            1010 – Configuration Read;            1011 – Configuration Write.            1100 – Memory Read Multiple;            1110 – Memory Read Line;            1111 – Memory Write and Invalidate.</p> <p>Эти разряды передается на выходы nCBE[3:0] в фазе адреса.            В фазе данных на этих выводах устанавливается значение 0x0</p>
0	RUN	<p>Режим работы PMSC:</p> <p>1 – выполняется процедура передачи блока данных;            0 – передача блока данных завершена или не запущена.</p> <p>Устанавливается в 1 при записи 1 в этот разряд.            Устанавливается в 0, если блок данных передан полностью или транзакция завершилась либо по фатальной ошибке, либо по программному останову</p>

#### 4.3.2.2 Адресные регистры обмена

32-разрядный регистр AR\_PCI предназначен для указания начального адреса на шине PCI при выполнении транзакции на шине PCI в режиме Master. При выполнении конфигурационных операций разряды AR\_PCI[1:0] определяют тип обмена (Type0 или Type1), а унитарный код в разрядах AR\_PCI[31:11] указывает IDSEL адресуемого устройства. Разряды AR\_PCI[10:2] должны быть установлены в соответствии со спецификацией Local Bus Specification Rev. 2.3 для адресуемого устройства

32-разрядный регистр IR\_MASTER хранит начальный адрес абонента микросхемы 1892XD1Я при выполнении транзакции на шине PCI в режиме Master.

Содержимое регистров AR\_PCI и IR\_MASTER в процессе обмена данными не изменяется.

#### 4.3.2.3 Регистр STATUS\_MASTER

Регистр STATUS\_MASTER предназначен для контроля передачи блока данных по шине PCI в режиме Master.

Формат регистра STATUS\_MASTER приведен в Таблица 4.6

**Таблица 4.6 Формат регистра STATUS\_MASTER**

Номер разряда	Условное обозначение	Назначение
31	Master Read Parity Error	<p>Ошибка при выполнении чтения на шине PCI.</p> <p>Устанавливается в 1, если при установленных битах Parity Error Response регистра Status/Command и Master Parity Stop регистра CSR_PCI обнаружена ошибка чётности</p>
30	Master Write Parity Error	<p>Ошибка при выполнении записи на шине PCI.</p> <p>Устанавливается в 1, если при установленных битах Parity Error Response регистра Status/Command и Master Parity Stop регистра CSR_PCI обнаружен низкий уровень сигнала nPERR</p>
29	Received Master Abort	Состояние признака Received Master Abort в регистре Status/Command

Номер разряда	Условное обозначение	Назначение
28	Received Target Abort	Состояние признака Received Target Abort в регистре Status/Command
27	No Trdy	Транзакция завершается установкой признака No Trdy при отсутствии сигналов nTRDY или nSTOP в течение времени "Master Initial Latency" после начала передачи
26	No Gnt	Признак отсутствия сигнала nGNT в течение 4095 тактов шины PCI после установки сигнала nREQ
25	Break Done	Состояние признака Break Done в регистре CSR_MASTER
24	Disconnect	Транзакция завершается установкой признака Disconnect при обнаружении ситуации nSTOP=0, nDEVSEL=0 в фазе передачи данных
23	Retry	Транзакция завершается установкой признака Retry при обнаружении ситуации nSTOP=0, nDEVSEL=0, nTRDY=1 в интервале времени "Master Initial Latency". Если Master Break=0, то данная транзакция будет повторяться до тех пор, пока она успешно не завершится и передача последующих данных продолжится. Если Master Break =1, то данная транзакция повторяться не будет, и передача данных прекратиться
22	Timeout	Транзакция завершается установкой признака Timeout, если при высоком уровне сигнала nGNT длительность транзакции (в тактах PCLK) превысила лимит, установленный разрядами MLT регистра Latency Timer
21	IRDYout	Транзакция завершается установкой признака IRDYout при отсутствии сигнала nIRDY в течение времени "IRDY Latency" после завершения очередной фазы передачи данных
20	TRDYout	Транзакция завершается установкой признака TRDYout при отсутствии сигналов nTRDY или nSTOP в течение времени "Master Subsequent Latency" после завершения очередной фазы данных
19:17	-	Не используется
16	RUN	Состояние признака RUN в регистре CSR_MASTER
15:0	WCC	Текущий размер блока данных: перед выполнением первой транзакции WCC = WC; после передачи по шине PCI очередного слова WCC = WCC – 1; после передачи последнего слова блока данных WCC = 0

Разряды 31:20 определяют причину окончания транзакции передачи данных в режиме Master. Эти разряды аппаратно обнуляются при запуске транзакции.

#### 4.3.2.4 Регистр параметров TMR\_PCI

32-разрядный регистр TMR\_PCI используется для хранения временных параметров выполнения транзакции на шине PCI в режиме Master.

Формат регистра TMR\_PCI приведен в Таблица 4.7

**Таблица 4.7 Формат регистра TMR\_PCI**

Номер разряда	Условное обозначение	Назначение
31:16	WaterMark	Пороговое значение для формирования прерывания MASTER_WMARK: прерывание формируется после передачи WaterMark слов по шине PCI 0x0 – 1слово, 0x1– 2слова , ..., 0xFFFF – 65536 слов;
15:12	-	Не используется
12: 8	MILat	Master Initial Latency в тактах PCLK

Номер разряда	Условное обозначение	Назначение
7:4	MSLat	Master Subsequent Latency в тактах PCLK
3: 0	IRDLat	nIRDY Latency в тактах PCLK

Если в регистре CSR\_MASTER sel\_MILat = 1 и MILat > 1, то используется MILat.

Если в регистре CSR\_MASTER sel\_MSLat =1, то используется MSLat.

Если в регистре CSR\_MASTER sel\_IRDLat =1, то используется IRDLat.

В противном случае используются значения, регламентированные в Local Bus Specification. Rev.2.3:

- Master Initial Latency = 0xF (16 тактов);
- Master Subsequent Latency = 7 (8 тактов);
- nIRDY Latency = 7(8 тактов).

Параметр nIRDY Latency определяет количество тактов ожидания (Waite State), которое может вставить PMSC при выполнении транзакции в режиме Master.

#### 4.3.2.5 Регистр CSR\_WIN

Регистр CSR\_WIN предназначен для задания параметров выполнения обменов с шиной PCI через адресное окно.

Формат регистра CSR\_WIN приведен в Таблица 4.8

**Таблица 4.8 Формат регистра CSR\_WIN**

Номер разряда	Условное обозначение	Назначение
31:24	AR_WIN	если SEL_ADR =1, то состояние этого поля передается на выходы AD[31:24] в фазе адреса
23:20	nCBE_WIN	если SEL_nCBE =1, то состояние этого поля передается на выходы nCBE[3:0] в фазе данных
19:17	-	Не используется
16	MASK_DPE	Разрешение формирования прерывания MASTER_ERROR по ошибке чётности: 1- разрешено; 0- запрещено
15:12	-	Не используется
12:8	MIL_WIN	Значение "Master Initial Latency" в тактах PCLK
7	SEL_MIL	Выбор параметра "Master Initial Latency": 0 – используется номинальное значение данного параметра; 1 – используется значение, определенное полем MIL_WIN этого регистра
6	SEL_ADR	Выбор адреса: 0 – на выводах AD[31:24] в фазе адреса данных устанавливается значение, определенное разрядами 31:24 регистра AR_PCI; 1 – на выводах AD[31:24] в фазе адреса данных устанавливается значение, определенное разрядами AR_WIN этого регистра
5	SEL_nCBE	Выбор nCBE: 0 – на выводах nCBE в фазе данных устанавливается нулевое значение; 1 – на выводах nCBE в фазе данных устанавливается значение, определенное разрядами nCBE_WIN этого регистра

Номер разряда	Условное обозначение	Назначение
4:1	CMD_WIN	Тип команды: 0xA или 0xB – выполняются команды типа Configuration; 0x2 или 0x3 – выполняются команды типа I/O; при остальных значениях данного поля выполняются команды Memory Write, Memory Read;
0	-	Не используется

#### 4.3.2.6 Регистр CSR\_PCI

Формат регистра CSR\_PCI приведен в Таблица 4.9

**Таблица 4.9** Формат регистра CSR\_PCI

Номер разряда	Условное обозначение	Назначение
31	Master Read Parity Error	Состояние признака Master Read Parity Error в регистре STATUS_MASTER
30	Master Write Parity Error	Состояние признака Master Write Parity Error в регистре STATUS_MASTER
29	Received Master Abort	Состояние признака Received Master Abort в регистре Status/Command
28	Received Target Abort	Состояние признака Received Target Abort в регистре Status/Command
27	No Trdy	Состояние признака No Trdy в регистре STATUS_MASTER
26	No Gnt	Состояние признака No Gnt в регистре STATUS_MASTER
25	Break Done	Состояние признака Break Done в регистре CSR_MASTER
24	Master Break	Программный останов передачи блока данных: 0 – разрешение выполнения транзакций передачи данных; 1 – запрещение выполнения транзакций передачи данных
23	mst_DPErd	Устанавливается в 1, если обнаружена ошибка чётности при выполнении чтения на шине PCI в режиме Master. Обнуляется при записи 1 в разряд RUN регистра CSR_MASTER. Доступ: только чтение
22	mst_DPEwr	Устанавливается в 1, если бит Parity Error Response регистра Status/Command установлен в 1 и обнаружен низкий уровень сигнала nPERR при выполнении записи на шине PCI в режиме Master. Обнуляется при записи 1 в разряд RUN регистра CSR_MASTER. Доступ: только чтение
21	-	Не используется
20	Master Parity Stop	Разрешение прекращения передачи блока данных и формирования прерывания MASTER_ERROR при обнаружении ошибки чётности в режиме Master.
19	tgt_DPEd	Устанавливается в 1, если обнаружена ошибка чётности в фазе данных при выполнении записи на шине PCI в режиме Target. Устанавливается в 0 при чтении этого регистра. Доступ: только чтение
18	tgt_DPEa	Устанавливается в 1 в режиме Target, если в фазе адреса обнаружена ошибка чётности. Устанавливается в 0 в режиме Target, если в фазе адреса ошибка чётности не обнаружена. Устанавливается в 0 при чтении этого регистра. Доступ: только чтение
17	-	Не используется

Номер разряда	Условное обозначение	Назначение
16	Target Parity Stop	Разрешение завершения транзакции в режиме Target установкой признака Signaled Target Abort при обнаружении ошибки чётности в фазе адреса
15:12	TILat	Target Initial Latency в тактах PCLK. Номинальное значение TILat = 0xC соответствует 16 тактам шины PCI
11:8	TSLat	Target Subsequent Latency в тактах PCLK. Номинальное значение TSLat = 0x6 соответствует 8 тактам шины PCI
7	Test par	Режим формирования выходного сигнала PAR: 0 – сигнал формируется в соответствии с Local Bus Specification Rev.2.3; 1 – формируется инверсное значение сигнала. Используется для тестирования PMSC
6	Test perr	Режим формирования выходного сигнала nPERR: 0 – сигнал формируется в соответствии с Local Bus Specification Rev.2.3; 1 – в режиме Target формируется инверсное значение сигнала. Используется для тестирования PMSC
5	sel_TILat	Разрешение изменения параметра "Target Initial Latency": 0 – используется номинальное значение данного параметра; 1 – используется значение, определенное полем TILat этого регистра
4	sel_TSLat	Разрешение изменения параметра "Target Subsequent Latency": 0 – используется номинальное значение данного параметра; 1 – используется значение, определенное полем TSLat этого регистра
3:0	WNT	Количество слов, которое должно накопиться в буфере WFIFO для передачи очередной порции данных в коммутатор SWITCH в режиме Target. Определяется из системных соображений, с точки зрения максимальной скорости передачи данных. Допустимые значения: 0, 7, 0xF. Если предполагаемая длина транзакции меньше или равна 16 слов, следует определить WNT = 0xF, в противном случае – выбрать WNT = 7 или WNT = 0.

Конфигурационная запись в этот регистр игнорируется.

Значение поля WNT используется при записи данных в память DPRAM в режиме Target: очередная порция данных в коммутатор SWITCH передается при накоплении в WFIFO не менее WNT слов. При WNT = 0, 1, 2, 3 передача в коммутатор SWITCH производится при накоплении в WFIFO не менее четырех слов.

При чтении данных из памяти DPRAM в режиме Target передача в шину PCI начинается по появлению первого слова в RFIFO.

#### 4.4 Обмен данными по шине PCI в режиме Target (Slave)

В режиме Target по шине PCI доступны следующие ресурсы микросхемы 1892XD1Я: регистры PMSC, регистры блоков SWIC, DMA\_SWIC и память DPRAM.

Регистры PMSC доступны по командам Configuration Read, Configuration Write в области адресов Type 0 и по командам Memory Read, Memory Write. Все транзакции обмена данными с регистрами PMSC выполняются за три такта шины PCI и завершаются после передачи первого слова установкой требования "Disconnect" (низкий уровень сигнала nSTOP).

Обмен данными с DPRAM и регистрами SWIC выполняется по командам Memory Read, Memory Write.

Команды Memory Read Multiple, Memory Read Line исполняются как Memory Read, а Memory Write and Invalidate – как Memory Write.

Обмен данными производится 32-разрядными словами (формат DWORD) независимо от состояния выводов nCBE в фазе данных.

Передача данных в режиме Target регламентируется битами управления и параметрами регистров Status/Command и CSR\_PCI и битом RUN регистра CSR\_MASTER.

Если бит Memory Space регистра Status/Command установлен в 0, то PMSC инициирует завершение команд Memory Read, Memory Write по условию “Master abort” установкой высокого уровня сигнала nDEVSEL. Состояние бита Memory Space не влияет на выполнение конфигурационных операций.

Если бит Target Parity Stop регистра CSR\_PCI установлен в 1, и обнаружена ошибка чётности в фазе адреса, то PMSC инициирует завершение обмена по условию “Target abort” установкой высокого уровня сигнала nDEVSEL и низкого уровня сигнала nSTOP.

Если бит RUN регистра CSR\_MASTER установлен в 1 (PMSC занят передачей блока данных в режиме Master), то PMSC как устройство “Target” инициирует завершение обмена данными с DPRAM и регистрами SWIC по условию “Retry” установкой высокого уровня сигнала nTRDY и низкого уровня сигнала nSTOP. Состояние бита RUN не влияет на выполнение операций с регистрами PMSC.

Если значение AD[1:0] в фазе адреса (Burst Order) отлично от нуля, то PMSC инициирует завершение обмена по условию “Disconnect” установкой низкого уровня сигнала nSTOP после передачи первого слова.

## 4.5 Обмен данными с шиной PCI в режиме Master

В режиме Master PMSC обеспечивает обмен данными между устройством на шине PCI и абонентами микросхемы 1892XD1Я: регистрами блоков SWIC, DMA\_SWIC и памятью DPRAM (см. Таблица 2.2).

В режиме Master на шине PCI могут выполняться команды: I/O Read, I/O Write, Memory Read, Memory Write, Configuration Read, Configuration Write, Memory Read Multiple, Memory Read Line, Memory Write and Invalidate. Код выполняемой команды определяется полем CMD регистра CSR\_MASTER.

Команды Memory Read Multiple и Memory Read Line выполняются как Memory Read, а команда Memory Write and Invalidate – как Memory Write. В зависимости от содержимого разрядов AR\_PCI[1:0] могут выполняться конфигурационные операции Type 0 и Type 1.

Передача данных регламентируется параметрами, установленными регистрами CSR, Latency Timer и регистрами управления обменом.

Запуск процедуры передачи осуществляется записью 1 в разряд 0 регистра CSR\_MASTER через адаптер MBA, или с шины PCI в режиме Target.

Для запуска с шины PCI необходимо предварительно установить в 1 бит Bus Master в регистре Status/Command. При запуске через адаптер MBA состояние бита Bus Master несущественно.

Перед запуском необходимо убедиться в том, что PMSC не выполняет передачу данных в режиме Master: в регистре CSR\_MASTER бит RUN установлен в 0.

Затем необходимо записать:

- начальный адрес абонента микросхемы 1892XD1Я в регистр IR\_MASTER;
- начальный адрес устройства на шине PCI в регистр AR\_PCI;
- параметры WaterMark, MlLat, MSLat, IRDLat в регистр TMR\_PCI;
- параметры WC, WNM, sel\_MlLat, sel\_MSLat, sel\_IRDLat, CMD и бит RUN=1 в регистр CSR\_MASTER.

После записи 1 в разряд RUN регистра CSR\_MASTER PMSC выполняет следующую процедуру передачи блока данных:

1. PMSC формирует запрос на шину PCI, устанавливая низкий уровень на выходе nREQ, и ожидает разрешения на занятие шины от арбитра шины (низкий уровень сигнала nGNT).
2. Если разрешение не поступило в течение 4095 тактов шины PCI, передача завершается по фатальной ошибке No Gnt:
  - в регистре CSR\_MASTER: DONE = 1, Fatal Error =1, RUN=0;
  - в регистре STATUS\_MASTER: No Gnt =1, RUN=0;
  - в регистре QSTR\_PCI: MASTER\_DONE=0, MASTER\_ERROR=1.
3. Если разрешение на занятие шины получено, PMSC снимает запрос nREQ и запускает выполнение транзакции на шине PCI установкой низкого уровня сигнала nFRAME.
4. Если после выполнения транзакции текущий размер блока данных (WCC) отличен от нуля, то PMSC продолжает передачу блока данных, формируя очередной запрос на шину PCI. Процедура передачи блока данных продолжается до тех пор, пока не будет передано последнее слово.
5. Если передача блока данных завершается по ошибке чётности (в регистре CSR\_PCI установлен бит Master Parity Stop, а в регистре STATUS\_MASTER установлен бит Master Read Parity Error или бит Master Write Parity Error), то при чтении передано с ошибкой только последнее слово, а при записи – последнее и, возможно, предпоследнее.
6. Если перед выполнением транзакции в регистре CSR\_PCI был установлен в 1 бит Master Break, то после ее выполнения передача данных прекращается. При этом, в регистре CSR\_MASTER: DONE=1, Fatal Error=0, Break Done=1, RUN=0;
7. Если транзакция завершается по фатальной ошибке (установлен в 1 один из разрядов STATUS\_MASTER [31:26]), то передача блока данных прекращается:
  - в регистре CSR\_MASTER: DONE=1, Fatal Error =1, RUN=0;
  - в регистре QSTR\_PCI: MASTER\_DONE=0, MASTER\_ERROR=1.
8. Если в результате выполнения транзакции передано последнее слово данных, то передача блока данных завершается:
  - в регистре CSR\_MASTER: DONE=1, Fatal Error=0, RUN=0;
  - в регистре QSTR\_PCI: MASTER\_DONE=1, MASTER\_ERROR=1 при mst\_DPErd =1 или mst\_DPEwr=1.

При обнаружении фатальной ошибки необходимо передать управление системной программе.

В промежутках между транзакциями по шине PCI доступны для чтения и записи только регистры контроллера PMSC. При обращении к регистрам SWIC или к DPRAM PMSC как устройство “Target” завершает транзакцию требованием повтора передачи “Retry” (nSTOP=0, nTRDY=1 в начальной фазе данных).

## 4.6 Обмен данными с шиной PCI через адресное окно

Внешний микропроцессор через адаптер MBA может обмениваться данными с шиной PCI по операциям записи и чтения. Для этого в адресном пространстве микросхемы

1892ХД1Я выделено адресное окно размером 16 Мбайт: диапазон адресов от 000\_0000 до 0FF\_FFFF. При чтении данных разряды 23:0 адреса устройства на шине PCI определяются разрядами [23:0] регистра BDR.

При записи данных разряды 23:0 адреса устройства на шине PCI определяются конкатенацией A[21:0] и двух нулей в младших разрядах.

Перед выполнением обмена данными необходимо убедиться, что PMSC не выполняет передачу данных в режиме Master: в регистре CSR\_MASTER бит RUN установлен в 0. Затем необходимо записать в регистр CSR\_WIN значения параметров SEL\_ADR, AR\_WIN, SEL\_nCBE, CBE\_WIN, SEL\_MIL, MIL\_WIN, CMD\_WIN, MASK\_DPE.

При обращении в окно шины PCI в регистре CSR\_MASTER аппаратно устанавливаются нулевые значения параметров WC, WNM, sel\_MSLat, sel\_IRDLat, единичные значения признаков RUN и WINDOW, и выполняется процедура передачи блока данных (см. п. 6.5).

При этом если CMD\_WIN = 0xA или 0xB, на шине PCI выполняется однословная команда Configuration Read или Configuration Write,

если CMD\_WIN = 0x2 или 0x3, выполняется однословная команда I/O Read или I/O Write,

иначе - команда Memory Read или Memory Write.

Если транзакция завершается по фатальной ошибке (установлен в 1 один из разрядов STATUS\_MASTER [31:26]), то

- в регистре CSR\_MASTER: DONE = 1, Fatal Error =1, Break Done =0, WINDOW=1; RUN=0;
- в регистре QSTR\_PCI: MASTER\_ERROR = 1, MASTER\_DONE =0.

Если транзакция завершается передачей данных то:

- STATUS\_MASTER [31:0] =0
- в регистре CSR\_MASTER: DONE = 1, Fatal Error =0, Break Done =0, WINDOW=1; RUN=0;
- в регистре QSTR\_PCI:

если обнаружена ошибка чётности и MASK\_DPE =1,

то MASTER\_ERROR= 1, MASTER\_DONE =1,

если ошибка чётности не обнаружена или замаскирована (MASK\_DPE =0),

то MASTER\_ERROR=0, MASTER\_DONE =0.

Если транзакция завершается по условию “Retry” (STATUS\_MASTER [23] =1) и установлен признак Master Break(CSR\_PCI[24] =1), то

- в регистре CSR\_MASTER: DONE = 1, Fatal Error =0, Break Done =1, WINDOW=1; RUN=0;
- в регистре QSTR\_PCI: MASTER\_ERROR = 1, MASTER\_DONE =1;
- в регистре STATUS\_MASTER: Break Done =1, Retry =1.

Если транзакция завершается по условию “Retry” (STATUS\_MASTER [23] =1) и признак Master Break сброшен (CSR\_PCI[24] =0), то PMSC повторяет транзакцию до тех пор, пока она не завершится передачей данных.

## 4.7 Передача прерываний

### 4.7.1 Передача вектора прерывания из шины PCI

Из шины PCI можно передать вектор прерывания внешнему процессору. Это выполняется с помощью регистров почтового ящика MBR\_PCI и семафора SEM.

32-разрядный регистр MBR\_PCI предназначен для хранения вектора прерывания. Разряд 0 регистра SEM является признаком занятости MBR\_PCI по записи со стороны шины PCI: при SEM = 0 регистр он свободен, а при SEM = 1 – занят. Разряды 31:1 регистра SEM не используются.

Перед записью в регистр MBR\_PCI со стороны шины PCI следует убедиться, что он свободен. Для этого необходимо опросить состояние семафора SEM командой Memory Read. После выполнения этой команды нулевой разряд регистра SEM аппаратно устанавливается в 1, поэтому при следующем чтении MBR\_PCI будет уже занят. Этот механизм позволяет избежать конфликта при совместном использовании регистра MBR\_PCI несколькими драйверами PCI.

При записи в регистр MBR\_PCI по команде Memory Write в регистре QSTR формируется прерывание INT\_MBR. Это прерывание сбрасывается при считывании содержимого MBR из MBA. После обработки прерывания признак занятости MBR\_PCI может быть сброшен записью нуля в регистр SEM из PCI по команде Memory Write или из MBA.

Конфигурационная запись в регистры MBR\_PCI и SEM не выполняется.

### 4.7.2 Передача прерываний в шину PCI

Все сигналы прерываний от SWIC и DMA SWIC поступают на регистр QSTR\_PCI. Формат регистра QSTR\_PCI аналогичен регистру QSTR (за исключением разряда 28) и приведен в Таблица 4.10

Из MBA можно передать вектор прерывания в шину PCI с помощью 32-разрядного регистра почтового ящика MBR\_MBA. При записи в этот регистр со стороны MBA в регистре QSTR\_PCI формируется прерывание INT\_MBA. Это прерывание сбрасывается при считывании содержимого MBR\_MBA по шине PCI.

**Таблица 4.10 Назначение разрядов регистра QSTR\_PCI**

Разряд регистра	Условное обозначение прерывания	Причина прерывания
Прерывания от PMSC		
31	MASTER_DONE	Окончание передачи блока данных в режиме Master. Обнуляется при чтении регистра QSTR_PCI
30	MASTER_ERROR	Ошибка при передаче блока данных в режиме Master. Обнуляется при чтении регистра QSTR_PCI
29	MASTER_WMARK	Предварительное уведомление о передаче по шине PCI в режиме Master заданного полем Water Mark регистра TMR_PCI количества слов. Обнуляется при чтении регистра QSTR_PCI
Прерывание от MBA		
28	INT_MBA	Признак записи данных в регистр почтового ящика MBR_MBA контроллера PMSC со стороны MBA. Обнуляется при чтении регистра MBR_MBA по шине PCI
Прерывания от DMA_SWIC3		
27	SWIC3_TX_DATA	Окончание передачи блока данных или цепочки блоков данных каналом DMA SWIC3_TX_DATA

Разряд регистра	Условное обозначение прерывания	Причина прерывания
26	SWIC3_TX_DESC	Окончание передачи блока данных или цепочки блоков данных каналом DMA SWIC3_TX_DESC
25	SWIC3_RX_DATA	Окончание передачи блока данных или цепочки блоков данных каналом DMA SWIC3_RX_DATA
24	SWIC3_RX_DESC	Окончание передачи блока данных или цепочки блоков данных каналом DMA SWIC3_RX_DESC
<u>Прерывания от DMA SWIC2</u>		
23	SWIC2_TX_DATA	Окончание передачи блока данных или цепочки блоков данных каналом DMA SWIC2_TX_DATA
22	SWIC2_TX_DESC	Окончание передачи блока данных или цепочки блоков данных каналом DMA SWIC2_TX_DESC
21	SWIC2_RX_DATA	Окончание передачи блока данных или цепочки блоков данных каналом DMA SWIC2_RX_DATA
20	SWIC2_RX_DESC	Окончание передачи блока данных или цепочки блоков данных каналом DMA SWIC2_RX_DESC
<u>Прерывания от DMA SWIC1</u>		
19	SWIC1_TX_DATA	Окончание передачи блока данных или цепочки блоков данных каналом DMA SWIC1_TX_DATA
18	SWIC1_TX_DESC	Окончание передачи блока данных или цепочки блоков данных каналом DMA SWIC1_TX_DESC
17	SWIC1_RX_DATA	Окончание передачи блока данных или цепочки блоков данных каналом DMA SWIC1_RX_DATA
16	SWIC1_RX_DESC	Окончание передачи блока данных или цепочки блоков данных каналом DMA SWIC1_RX_DESC
<u>Прерывания от DMA SWIC0</u>		
15	SWIC0_TX_DATA	Окончание передачи блока данных или цепочки блоков данных каналом DMA SWIC0_TX_DATA
14	SWIC0_TX_DESC	Окончание передачи блока данных или цепочки блоков данных каналом DMA SWIC0_TX_DESC
13	SWIC0_RX_DATA	Окончание передачи блока данных или цепочки блоков данных каналом DMA SWIC0_RX_DATA
12	SWIC0_RX_DESC	Окончание передачи блока данных или цепочки блоков данных каналом DMA SWIC0_RX_DESC
<u>Прерывания от SWIC3</u>		
11	SWIC3_TIME	Получен управляющий код
10	SWIC3_ERR	Ошибка в канале приема
9	SWIC3_LINK	Установлено соединение
<u>Прерывания от SWIC2</u>		
8	SWIC2_TIME	Получен управляющий код
7	SWIC2_ERR	Ошибка в канале приема
6	SWIC2_LINK	Установлено соединение
<u>Прерывания от SWIC1</u>		
5	SWIC1_TIME	Получен управляющий код
4	SWIC1_ERR	Ошибка в канале приема
3	SWIC1_LINK	Установлено соединение
<u>Прерывания от SWIC0</u>		
2	SWIC0_TIME	Получен управляющий код
1	SWIC0_ERR	Ошибка в канале приема
0	SWIC0_LINK	Установлено соединение

Регистр QSTR\_PCI доступен только по чтению. Исходное состояние регистра QSTR\_PCI - нули (нет запросов прерывания).

Каждое прерывание, поступающее на регистр QSTR\_PCI, маскируется при помощи регистра маски MASKR\_PCI, формат которого аналогичен формату регистра QSTR\_PCI. Регистр MASKR\_PCI доступен по записи и чтению. Исходное состояние регистра MASKR\_PCI - нули (все прерывания замаскированы).

Все незамаскированные прерывания регистра QSTR\_PCI объединяются по схеме ИЛИ, и результат поступает на вход Interrupt Status регистра Status/Command.

При Interrupt Desable = 0 (бит 10 регистра Status/Command) и Interrupt Status =1 на внешнем выводе nINTA устанавливается низкий уровень сигнала. В противном случае - пассивный уровень.

В процессе обслуживания прерывания необходимо проанализировать состояние устройства для определения причины его возникновения. Сброс прерывания осуществляется в момент исключения причины возникновения данного прерывания. Например, прерывание SWIC1\_LINK сбрасывается при записи 1 в разряд [12] регистра статуса SWIC1\_STATUS.

## 4.8 Арбитр

Контроллер PMSC содержит арбитр шины PCI, имеющий 5 входов nREQ[4:0] запроса доступа к шине PCI и 5 выходов разрешения доступа nGNT[4:0].

В арбитра реализована одноуровневая схема приоритета доступа к шине PCI. Взаимный приоритет запросов nREQ[4:0] изменяется циклически в соответствии с Таблица 4.11 после каждого предоставления шины PCI очередному мастеру. Исходное распределение приоритетов между запросами ( в порядке их убывания ):

nREQ[0], nREQ[1], nREQ[2], nREQ[3], nREQ[4].

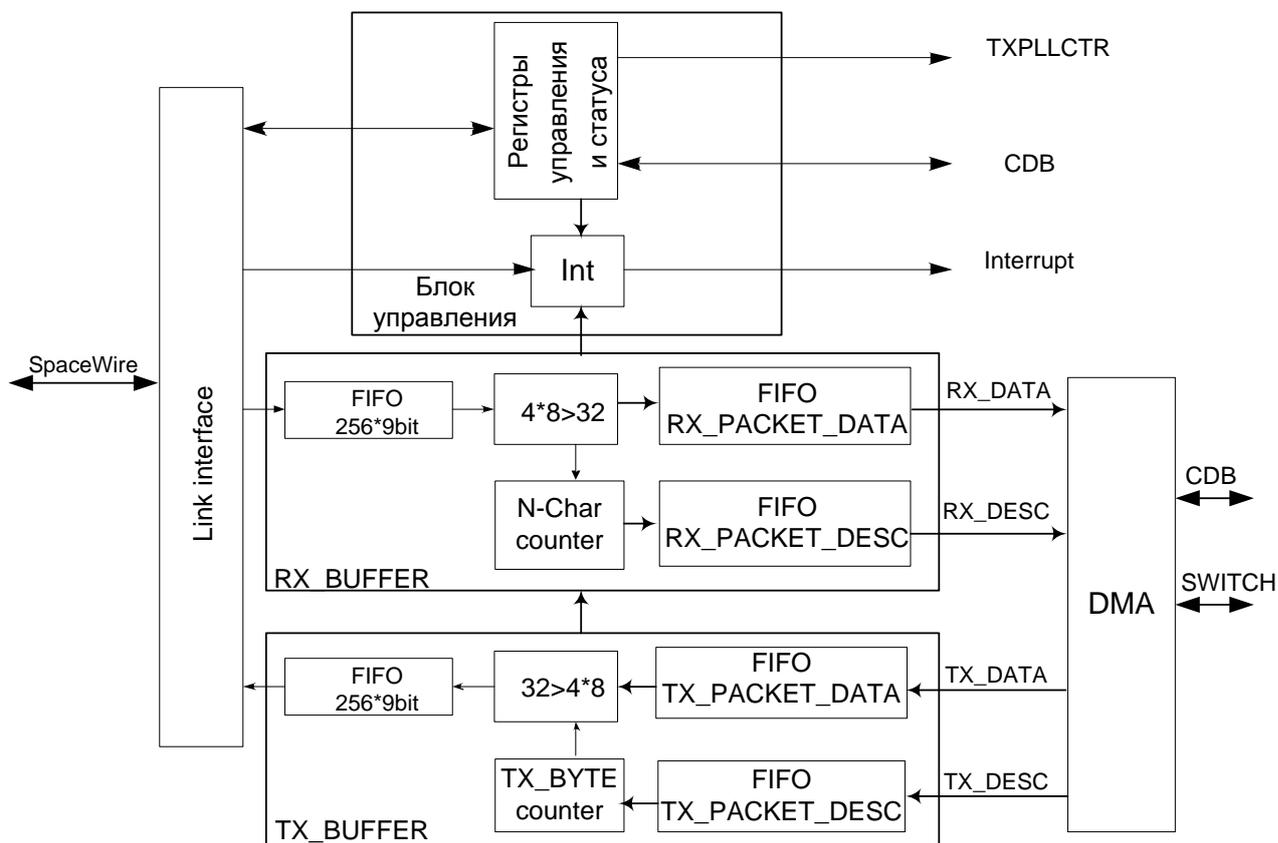
**Таблица 4.11 Приоритеты арбитра шины PCI**

Обслуживаемый запрос	Распределение приоритетов очередного обмен
nREQ[0]	nREQ[1], nREQ[2], nREQ[3], nREQ[4], nREQ[0]
nREQ[1]	nREQ[2], nREQ[3], nREQ[4], nREQ[0], nREQ[1]
nREQ[2]	nREQ[3], nREQ[4], nREQ[0], nREQ[1], nREQ[2]
nREQ[3]	nREQ[4], nREQ[0], nREQ[1], nREQ[2], nREQ[3]
nREQ[4]	nREQ[0], nREQ[1], nREQ[2], nREQ[3], nREQ[4]

## 5. КОНТРОЛЛЕР SWIC

### 5.1 Структурная схема

Структурная схема SWIC приведена на Рисунок 5.1.



**Рисунок 5.1 Структурная схема SWIC**

Основой SWIC является блок Link interface (см. Figure 19 стандарта ECSS-E-50-12A), который через драйверы LVDS подключен к физическим линиям канала связи SpaceWire.

Управление SWIC осуществляется внешним микропроцессором или PMSC. Внешний микропроцессор (через MBA) или Master шины PCI (через PMSC) могут выполнять чтение и запись регистров контроллера для определения его состояния и настройки параметров работы.

Блок управления задает режимы работы блока Link interface. Передача управляющих кодов, кода распределенного прерывания, roll кода и контроль состояния последнего полученного извне маркера времени, производится через соответствующие регистры блока. В этом блоке находятся также регистры коэффициента скорости передачи данных и коэффициента скорости приема данных.

Блок формирования прерываний Int формирует соответствующие прерывания по состоянию блока Link interface.

Обмен данными между DPRAM и SWIC выполняется при помощи 4-канального DMA:

- канал дескрипторов передаваемых пакетов;
- канал данных передаваемых пакетов;
- канал дескрипторов принимаемых пакетов;
- канал данных принимаемых пакетов.

Описания работы блоков DMA приведено в п. 5.5.

К DMA SWIC подключен через буферы приема данных RX\_BUFFER и передачи данных TX\_BUFFER. Они обеспечивают согласование скоростей передачи данных между каналом SpaceWire и каналами DMA .

Буфер приема RX\_BUFFER имеет конвейерную организацию и состоит из двух ступеней. Сначала в FIFO\_256\*9bit буферизируются восьмиразрядные данные, принимаемые от блока Link interface. Девятый служебный разряд несет информацию о признаке символа данных N-Char или символе конца пакета EOP. Затем в блоке преобразования формируются 32-разрядные слова данных и поступают в FIFO RX\_PACKET\_DATA. Дескриптор пакета формируется в счетчике N-Char\_counter. При поступлении символа данных N-Char счетчик увеличивается на 1, при поступлении символа конца пакета значение счетчика переписывается в выходной буфер RX\_PACKET\_DESC, а сам счетчик сбрасывается в 0.

В буфер передачи TX\_BUFFER с помощью канала передаваемых данных DMA записываются 32-разрядные слова данных. Содержимое пакетов и их дескрипторы буферизируются в двух FIFO TX\_PACKET\_DATA и TX\_PACKET\_DESC соответственно. Данные из буфера передачи в блок Link interface выдаются побайтно через FIFO 256\*9bit. Преобразование 32-хразрядных слов в байты осуществляется в блоке преобразования под управлением счетчика TX\_BYTE counter. В счетчик заносится размер пакета из дескриптора передаваемого пакета. После передачи каждого байта этот счетчик уменьшается на 1. По достижении счетчиком значения 0, в поток передаваемых данных вставляется символ конца пакета EOP, а в счетчик заносится размер следующего передаваемого пакета из следующего дескриптора.

## 5.2 Программная модель

### 5.2.1 Общие положения

В состав микросхемы 1892XD1Я входят 4 контроллера SWIC: SWIC0, SWIC1, SWIC2, SWIC3. Выбор контроллера осуществляется по базовому адресу. Соответствие базовых адресов контроллерам приведено в Таблица 5.1. Выбор регистра внутри адресного пространства контроллера осуществляется битами адреса с 5 по 2. Младшие два бита адреса устанавливаются в 0.

**Таблица 5.1 Базовые адреса контроллеров SWIC**

Контроллер	Базовый адрес
SWIC0	0x140_0000
SWIC1	0x160_0000
SWIC2	0x180_0000
SWIC3	0x1A0_0000

Перечень программно-доступных регистров SWIC приведен в Таблица 5.2.

**Таблица 5.2 Регистры SWIC**

Условное обозначение регистра	Название регистра	Тип доступа
-------------------------------	-------------------	-------------

Условное обозначение регистра	Название регистра	Тип доступа
HW_VER	Номер версии контроллера	RD
STATUS	Регистр состояния	WR/RD
RX_CODE	Регистр управляющего символа, принятого из сети (маркера времени, кода распределенного прерывания или кода подтверждения распределенного прерывания)	RD
MODE_CR	Регистр режима работы	WR
TX_SPEED	Регистр коэффициента скорости передачи	WR
TX_CODE	Регистр управляющего символа (маркера времени, кода распределенного прерывания, кода подтверждения) для передачи в сеть	WR
RX_SPEED	Регистр скорости приема данных в канале SpaceWire.	RD
CNT_RX_PACK	Регистр счетчика принятых пакетов ненулевой длины	RD/WR
CNT_RX0_PACK	Регистр счетчика принятых пакетов нулевой длины (идущих подряд символов концов пакетов)	RD/WR
ISR_L	Младшие разряды регистра ISR	RD/WR
ISR_H	Старшие разряды регистра ISR	RD/WR
TRUE_TIME	Регистр достоверного маркера времени	RD
TOUT_CODE	Регистр размера таймаутов	RD/WR
ISR_tout_L	Младшие разряды регистра флагов таймаутов ISR	RD/WR
ISR_tout_H	Старшие разряды регистра флагов таймаутов ISR	RD/WR
LOG_ADDR	Регистр логического адреса	RD/WR

Обмен данными с регистрами осуществляется 32-разрядными словами. Если разряды регистра не используются, то из них считываются нули. При записи в этих разрядах необходимо указывать нули.

В данном разделе используются следующие обозначения:

- R – доступен только по чтению;
- W – доступен только по записи;
- RW – доступен по чтению и записи;
- RW1 – доступен по чтению и записи 1. Запись 1 приводит к установке данного разряда в 0.

## 5.3 Описание регистров SWIC

### 5.3.1 Регистр HW\_VER

Регистр номера версии SWIC. При чтении этого регистра выводится номер версии аппаратной реализации SWIC – “0x0000 0003”.

**Таблица 5.3 Назначение разрядов регистра HW\_VER**

Номер разряда	Условное обозначение	Описание
31:0	HW_VER	Номер версии SWIC

### 5.3.2 Регистр STATUS

Регистр состояния блока SWIC предназначен для оперативного контроля состояния фаз работы контроллера. Регистр доступен как на чтение, так и на запись. Заполнение регистра выполняется побитно по сигналам от Link-интерфейса, блока приема данных из канала SpaceWire, блока передачи данных в канал SpaceWire. Назначение разрядов регистра приведено в Таблица 5.4.

**Таблица 5.4 Назначение разрядов регистра STATUS**

Номер разряда	Условное обозначение	Описание
0	DC_ERR	Признак ошибки рассоединения (DisconnectError): «1» – Ошибка произошла «0» – Нет ошибки (после сигнала сброса) Запись «1» в этот разряд сбрасывает этот разряд в «0». Используется для сброса прерывания ERR посредством записи «1» в этот разряд. Исходное состояние - «0»
1	P_ERR	Признак ошибки четности: «1» – Ошибка произошла «0» – Нет ошибки (после сигнала сброса) Запись «1» в этот разряд сбрасывает этот разряд в "0". Используется для сброса прерывания ERR посредством записи «1» в этот разряд. Исходное состояние - «0»
2	ESC_ERR	Признак ошибки в ESC последовательности: «1» – Ошибка произошла «0» – Нет ошибки (после сигнала сброса) Запись «1» в этот разряд сбрасывает этот разряд в «0». Используется для сброса прерывания ERR посредством записи «1» в этот разряд. Исходное состояние - «0»
3	CREDIT_ERR	Признак ошибки кредитования: «1» – Ошибка произошла «0» – Нет ошибки (после сигнала сброса) Запись «1» в этот разряд сбрасывает этот разряд в «0». Используется для сброса прерывания ERR посредством записи «1» в этот разряд. Исходное состояние - «0»
4	-	Не используется

Номер разряда	Условное обозначение	Описание
5 - 7	LINK_STATE	Состояние блока Link interface (см. Figure. 22 стандарта ECSS-E-50-12A): 000 - Error Reset (исходное состояние); 001 - Error Wait; 010 – Ready; 011 – Started; 100 – Connecting; 101 – Run.
8	RX_BUF_FULL	Состояние буфера приема: «1» – буфер полон; «0» – в буфере есть свободное место (после сигнала сброса). Исходное состояние - «0»
9	RX_BUF_EMPTY	Буфер приема пуст «1» – Пуст (после сигнала сброса) «0» – В буфере есть данные Исходное состояние - «1»
10	TX_BUF_FULL	Состояние буфера передачи: «1» – буфер полон; «0» – в буфере есть свободное место (после сигнала сброса). Исходное состояние - «0»
11	TX_BUF_EMPTY	Буфер передачи пуст «1» – Пуст (после сигнала сброса) «0» – В буфере есть данные. Исходное состояние - «1»
12	GOT_FIRST_BIT	Состояние принятого первого бита из канала «1» – бит принят «0» – приемный канал не активен (не было изменений фронтов din/sin после последнего сброса макроячейки по сбросу или в связи с ошибкой) Запись «1» в этот бит сбрасывает прерывание LINK, если оно было установлено, но не изменяет состояние GOT_FIRST_BIT. Исходное состояние - «0»
13	CONNECTED	Соединение установлено (LINK_STATE=5). Исходное состояние - «0»
14	GOT_TIME	Принят маркер времени из сети «1» – Принят маркер времени «0» – Маркер времени не принят (после сигнала сброса) Запись «1» в этот разряд сбрасывает этот разряд в «0». Используется для сброса прерывания TIME посредством записи «1» в этот разряд. Исходное состояние - «0»
15	GOT_INT	Принят код распределенного прерывания из сети «1» – Принят код распределенного прерывания времени «0» – Код распределенного прерывания не принят (после сигнала сброса) Запись «1» в этот разряд сбрасывает этот разряд в «0». Используется для сброса прерывания TIME посредством записи 1 в этот разряд. Исходное состояние «0».

Номер разряда	Условное обозначение	Описание
16	GOT_ACK	Принят код подтверждения из сети «1» – Принят код подтверждения «0» – код подтверждения не принят (после сигнала сброса) Запись «1» в этот разряд сбрасывает этот разряд в «0». Используется для сброса прерывания TIME посредством записи «1» в этот разряд. Исходное состояние - «0»
17	FL_CONTROL	Если данный флаг сброшен в «0», SWIC готов к отправке управляющего кода (маркера времени, кода распределенного прерывания, кода подтверждения). Если управляющий код записывается в SWIC при установленном флаге, его передача в сеть не гарантируется. Исходное состояние - «0»
18	LINK	Признак прерывания LINK (соединение установлено: разряд CONNECTED=1). Устанавливается, если оно не замаскировано в регистре режима. Сбрасывается при сбросе соответствующего прерывания. Исходное состояние «0».
19	ERR	Признак прерывания ERR (обнаружена ошибка в канале: имеет единичное состояние один из разрядов DC_ERR, P_ERR, ESC_ERR, CREDIT_ERR). Устанавливается, если оно не замаскировано в регистре режима. Сбрасывается при сбросе соответствующего прерывания. Исходное состояние «0».
20	TIME	Признак прерывания TIME (получен управляющий код: имеет единичное состояние один из разрядов GOT_TIME, GOT_INT, GOT_ACK, CC_01, CC_11 или истекло время таймаута приема кода распределенного прерывания). Устанавливается, если оно не замаскировано в регистре режима. Сбрасывается при сбросе соответствующего прерывания. Исходное состояние «0».
21	CC_11	Признак принятия управляющего кода C[7..6]=11 «1» – Принят управляющий код «0» – управляющий код не принят (после сигнала сброса) Запись «1» в этот разряд сбрасывает этот разряд в «0». Исходное состояние - «0»
22	CC_01	Признак принятия управляющего кода C[7..6]=01 «1» – Принят управляющий код «0» – Упр. код не принят (после сигнала сброса) Запись «1» в этот разряд сбрасывает этот разряд в «0». Исходное состояние - «0»
23..29	-	Резерв
30	S_LVDS_RX	Значение линии LVDS Sin при MODE_CR[29]=1.
31	D_LVDS_RX	Значение линии LVDS Din при MODE_CR[29]=1

### 5.3.3 Регистр RX\_CODE

Регистр принятого из сети управляющего кода. Назначение разрядов регистра приведено в Таблица 5.5. Исходное состояние регистра не определено.

**Таблица 5.5 Назначение разрядов регистра RX\_CODE**

Номер разряда	Условное обозначение	Описание
7:0	TIME_CODE	Значение маркера времени, принятого из сети последним
15:8	INT_CODE	Значение кода распределенного прерывания, принятого из сети последним
23:16	ACK_CODE	Значение кода подтверждения, принятого из сети последним
31:24	CC_11	Значение кода C[7..6]=11 принятого из сети последним

### 5.3.4 Регистр MODE\_CR

Регистр режима работы. Назначение разрядов регистра приведено в Таблица 5.6.

**Таблица 5.6 Назначение разрядов регистра MODE\_CR**

Номер разряда	Условное обозначение	Назначение
0	LinkDisabled	Установка LinkDisabled для блока Link interface
1	AutoStart	Установка Autostart для блока Link interface
2	LinkStart	Установка LinkStart для блока Link interface
3	-	Не используется
4	-	Не используется
5	-	Должен быть установлен в 0
6	RDY_MODE	Режим формирования признака готовности обмена данными с DMA SWIC: 0 – штатный режим работы. Признак готовности SWIC формирует аппаратно; 1 – признак готовности установлен в 1. Используется для приведение DMA SWIC в исходное состояние, если: произошло разъединение; необходимо программно остановить SWIC и его DMA
7	-	Не используется
8	TEST_TYPE	Тип режима работы ('0' – рабочий, '1' – тестовый)
9..10	-	Должны быть установлены в 0
11	LVDS_Loopback	Loopback (перед LVDS)
12	CODEC_Loopback	Loopback (перед кодеком)
13	LINK_Loopback	Loopback (перед Link interface)
14	COEFF_10_wr	Разрешение модификации регистра коэффициента для подсчета таймаутов
15	AUTO_SPEED	Если этот бит установлен в 1, то при разрыве соединения коэффициент скорости передачи будет автоматически устанавливаться на 10МГц, а при установке соединения автоматически переходит на базовое значение скорости
16	dIRQ_mode	Режим передачи/приема кодов распределенных прерываний. Если этот бит установлен в 0, то используются 6-и битные коды распределенных прерываний, если в 1 – то используются 5-и битные коды распределенных прерываний
17	-	Не используется

Номер разряда	Условное обозначение	Назначение
18	LINK_mask	Маска прерывания LINK. Если значение маски установлено в 1, то значение прерывания отображается в регистрах STATUS, QSTR, QSTR_PCI. Если значение 0, значение прерывания не отображается в этих регистрах
19	ERR_mask	Маска прерывания ERR. Если значение маски установлено в 1, то значение прерывания отображается в регистрах STATUS, QSTR, QSTR_PCI. Если значение 0, значение прерывания не отображается в этих регистрах
20	TIME_mask	Маска прерывания TIME. Если значение маски установлено в 1, то значение прерывания отображается в регистрах STATUS, QSTR, QSTR_PCI. Если значение 0, значение прерывания не отображается в этих регистрах
21	CTR	Если этот бит установлен в 1, то установка соединения выполняется без ожидания таймаутов (используется в отладочном режиме)
22	TCode_mask	Если этот разряд установлен в 0, то прерывание TIME при получении тайм-кода не устанавливается
23	INT_mask	Если этот разряд установлен в 0, то прерывание TIME при получении кода распределенного прерывания или кода подтверждения не устанавливается
24	CC_11_mask	Если этот разряд установлен в 0, то прерывание TIME при получении управляющего кода C[7..6]=11 не устанавливается
25	CC_01_mask	Если этот разряд установлен в 0, то прерывание TIME при получении управляющего кода C[7..6]=01 (dIRQ_regime=1) не устанавливается
26	INT_tout_mask	Если этот разряд установлен в 0, то прерывание TIME по факту истечения таймаута получения кода подтверждения не устанавливается
28:27	INT_tout_allow	Разрешение контроля таймаутов получения кодов подтверждения: 00 – контроль таймаутов запрещен 01 – выполняется контроль таймаутов и установка флагов истечения таймаутов 10 – выполняется контроль таймаутов, установка флагов истечения таймаутов и отправка кода подтверждения в сеть
29	LVDS_mode	Режим LVDS – если этот бит установлен в 0 – штатный режим работы, на выходные драйверы LVDS подаются сигналы от передатчика, разряды регистра STATUS[31:30] равны «0»; 1 – тестовый режим работы, на Sout, Dout LVDS подаются значения из разрядов 30, 31 регистра MODE_CR, в регистр STATUS[31:30] отображаются входные линии Sin и Din LVDS.
30	S_LVDS_TX	Значение для передачи на линию Sout LVDS
31	D_LVDS_TX	Значение для передачи на линию Dout LVDS

После того, как в результате разрешения AutoStart или LinkStart блок Link-интерфейса установил соединение (при LinkDisabled='0'), буфер передачи в сеть начинает принимать данные из DMA. Если DMA передал все данные, то далее в сеть передаются символы NULL. Соединение при этом не прекращается. Соединение прекращается, если процессор осуществляет запись единицы в бит LinkDisabled.

### 5.3.5 Регистр TX\_SPEED

Регистр коэффициентов скорости передачи. В разряды 9:0 записывается коэффициент, который передается на TXPLL при полностью программном управлении скоростью передачи. При использовании автоматического перехода на 10 МГц при разрыве соединения, коэффициент, записанный в разряды 9:0 устанавливается, когда соединение установлено. При разрыве соединения в этом режиме автоматически устанавливается коэффициент, записанный в разряды 19:10, он должен соответствовать скорости передачи 10 МГц.

В разряды 28:20 этого регистра записывается значение коэффициента для подсчета таймаутов установки соединения (6,4 мкс и 12,8 мкс)..

Запись нового значения в этот регистр возможно только, если бит COEFF\_10\_wt (14) регистра MODE\_CR (режима) установлен в 1.

**Таблица 5.7 Назначение разрядов регистра TX\_SPEED**

Номер разряда	Условное обозначение	Назначение
7:0	TX_SPEED	Определяет скорость передачи данных (в режиме авто установки скорости используется как базовое значение после установки соединения), Мбит/с: 0x01 - 5; 0x02 – 10; .... 0x4F – 395; 0x50 – 400.
8	PLL_TX_EN	Управление работой PLL_TX_SWIC: 1 – работа разрешена; 0 – работа запрещена. PLL_TX_SWIC находится в режиме пониженного энергопотребления
9	LVDS_EN	Управление работой приемопередатчиков LVDS SWIC: 1 – работа разрешена; 0 – работа запрещена. LVDS SWIC находятся в режиме пониженного энергопотребления
19:10	TX_SPEED_10	Определяет скорость передачи данных при установке соединения (в режиме авто установки скорости). Должен быть записан код 0x02
28:20	COEFF_10	Значение коэффициента для подсчета таймаутов установки соединения. Необходимо установить значение 0x0A
31..29	-	Резерв

### 5.3.6 Регистр TX\_CODE

Регистр управляющего кода для передачи в канал. Сразу же после записи в этот регистр начинается передача управляющего символа в Link-интерфейс и далее в канал.

**Таблица 5.8 Назначение разрядов регистра TX\_CODE**

Номер разряда	Условное обозначение	Описание
5:0	CODE_VAL	Значение управляющего кода для отправки в сеть
7:6	CODE_TYPE	Тип управляющего кода для отправки в сеть: 00 – код времени; 01 – код прерывания; 10 – код подтверждения прерывания.
31:8	-	Резерв

### 5.3.7 Регистр RX\_SPEED

Назначение разрядов регистра RX\_SPEED приведено в Таблица 5.9.

**Таблица 5.9 Назначение разрядов регистра RX\_SPEED**

Номер разряда	Условное обозначение	Назначение
31:8	-	Не используется

7:0	RX_SPEED	Скорость приема данных из канала связи SpaceWire равна (RX_SPEED*800/1024) Мбит/с. Например, если RX_SPEED=128 (десятичное), то скорость приема данных равна 100 Мбит/с
-----	----------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Исходное состояние регистра – нули.

### 5.3.8 Регистр CNT\_RX\_PACK

Регистр счетчика принятых пакетов. Значение регистра увеличивается на 1 каждый раз, когда из Link-интерфейса прочитывается символ конца пакета, если ему предшествовал один или более символ данных. Исходное состояние регистра «0».

При записи, значение регистра обнуляется. Процессор может обнулить содержимое этого регистра для того, чтобы начать счет пакетов заново. Рекомендуется выполнять сброс регистра каждый раз при выполнении новой настройки DMA для передачи данных в сеть.

**Таблица 5.10 Назначение разрядов регистра CNT\_RX\_PACK**

Номер разряда	Условное обозначение	Описание
31:0	CNT	Число принятых пакетов

### 5.3.9 Регистр CNT\_RX0\_PACK

Регистр счетчика принятых пустых пакетов. Значение регистра увеличивается на 1 каждый раз, когда из Link-интерфейса прочитывается символ конца пакета, если ему не предшествовал хотя бы один символ данных. Исходное состояние регистра «0».

При записи, значение регистра обнуляется. Процессор может обнулить содержимое этого регистра для того, чтобы начать счет пакетов заново. Рекомендуется выполнять сброс регистра каждый раз при выполнении новой настройки DMA для передачи данных в сеть.

**Таблица 5.11 Назначение разрядов регистра CNT\_RX0\_PACK**

Номер разряда	Условное обозначение	Описание
31:0	CNT	Число принятых пустых пакетов

### 5.3.10 Регистр ISR\_L

В этот регистр отображается младшая (31..0) часть регистра ISR. Регистр ISR содержит информацию о принятых и отправленных кодах распределенных прерываний и подтверждения. Если из сети получено распределенное прерывание, то бит регистра ISR, соответствующий номеру распределенного прерывания устанавливается в 1 (если он уже не был установлен в 1). Аналогично, если в регистр TX\_CODE осуществляется запись кода распределенного прерывания, соответствующий бит регистра ISR устанавливается в 1.

Если из сети получен код подтверждения, то бит регистра ISR, соответствующий номеру кода подтверждения устанавливается в 0 (если он уже не был установлен в 0). Аналогично, если в регистр TX\_CODE осуществляется запись кода подтверждения, соответствующий бит регистра ISR устанавливается в 0.

Необходимость данного регистра связана с тем, что коды распределенных прерываний и коды подтверждения могут приходиться из сети очень часто, быстрее, чем процессор может среагировать на очередное прерывание и прочитать код. Если даже в регистре RX\_CODE

код распределенного прерывания или код подтверждения будет перезаписан следующим, информация о нем не будет утрачена – она сохранится в регистре ISR.

Существует возможность программного сброса отдельных битов ISR. Для этого необходимо записать в соответствующие биты 1. (Если в бит записывается значение 0, то его значение не меняется).

**Таблица 5.12 Назначение разрядов регистра ISR\_L**

Номер разряда	Условное обозначение	Описание
31:0	ISR_L	Младшая часть регистра ISR

### 5.3.11 Регистр ISR\_H

В этот регистр отображается старшая [63:32] часть регистра ISR.

**Таблица 5.13 Назначение разрядов регистра ISR\_H**

Номер разряда	Условное обозначение	Описание
31:0	ISR_H	Старшая часть регистра ISR

### 5.3.12 Регистр TRUE\_TIME

В этот регистр записывается значение последнего правильного маркера времени, в отличие от разрядов 5:0 регистра RX\_CODE, в котором регистрируются все принятые маркеры времени. Назначение разрядов регистра приведено в Таблица 5.14. Исходное состояние регистра «0».

**Таблица 5.14 Назначение разрядов регистра TRUE\_TIME**

Номер разряда	Условное обозначение	Описание
5:0	TRUE_TIME	Значение последнего правильного маркера времени
31:6	Не используется	

### 5.3.13 Регистр TOUT\_CODE

В этот регистр записываются значение периода для глобального счетчика таймаутов (в количестве тактов локальной частоты) и максимальные значения локальных счетчиков таймаутов ожидания кодов подтверждения распределенных прерываний. Отдельный локальный счетчик таймаутов соответствует каждому разряду ISR. Если в SWIC поступает код распределенного прерывания, то запускается соответствующий ему счетчик локальных таймаутов. Он декрементируется каждый раз при завершении очередного периода счета глобального счетчика таймаутов.

**Таблица 5.15 Назначение разрядов регистра TOUT\_CODE**

Номер разряда	Условное обозначение	Описание
15..0	GLOB_COU	Значение периода глобального счетчика (задается в тактах локальной частоты)
20..16	LOC_COU1	Значение таймаута ожидания кода подтверждения (на код прерывания, отправленный процессором через SWIC)

Номер разряда	Условное обозначение	Описание
25..21	LOC_COU2	Значение таймаута ожидания кода подтверждения (на код прерывания, принятый из сети)
31:26	Не используется	

### 5.3.14 Регистр ISR\_tout\_L

В этот регистр отображается младшая (31..0) часть регистра флагов ISR\_tout. Если в регистре ISR регистрируется код распределенного прерывания, то для него запускается счет таймаута (каждому разряду ISR соответствует отдельный счетчик). В зависимости от того, был ли код распределенного прерывания принят из сети или отправлен процессором, начальное значение счетчика устанавливается в LOC\_TOUT1 или LOC\_TOUT2. (значение счетчика декрементируется каждый раз, когда глобальный счетчик досчитывает до определенного для него максимального значения). Если за время счета из сети не поступает соответствующий код подтверждения, то соответствующий разряд регистра ISR\_tout устанавливается в 1. Для того чтобы его сбросить, необходимо записать в этот разряд регистра ISR\_tout 1. (При записи в бит значения 0, его значение не меняется)

**Таблица 5.16 Назначение разрядов регистра ISR\_tout\_L**

Номер разряда	Условное обозначение	Описание
31:0	ISR_tout_L	Младшая часть регистра ISR_tout

### 5.3.15 Регистр ISR\_tout\_H

В этот регистр отображается старшая (63..32) часть регистра ISR\_tout.

**Таблица 5.17 Назначение разрядов регистра ISR\_tout\_H**

Номер разряда	Условное обозначение	Описание
31:0	ISR_tout_H	Старшая часть регистра ISR_tout

После сброса содержимое регистров «0».

### 5.3.16 Регистр LOG\_ADDR

В этом регистре хранится значение логического адреса, добавляемого к пакету по умолчанию, если установлен соответствующий режим. Длина логического адреса может быть от одного до 4 байтов, она определяется значением дескриптора пакета.

**Таблица 5.18 Назначение разрядов регистра LOG\_ADDR**

Номер разряда	Условное обозначение	Описание
31:0	LOG_ADDR	Значение логического адреса.

## 5.4 Логика работы SWIC

### 5.4.1 Установление соединения

Перед началом работы с контроллером SWIC необходимо при помощи регистра TX\_SPEED разрешить работу PLL\_TX и установить скорость передачи данных в канал связи SpaceWire величиной 10 Мбит/с. В соответствии со стандартом SpaceWire установление соединения следует производить на этой скорости. Для выхода PLL\_TX на рабочий режим требуется время не более 20 мс.

Для разрешения процесса установки соединения необходимо в регистре MODE\_CR установить: LinkDisabled =0, LinkStart =1.

Критерием успешного установления соединения является прохождение прерывания INT\_LINK и отсутствие прерывания INT\_ERR.

После обнаружения прерывания INT\_LINK, необходимо считать регистр STATUS и проверить биты DC\_ERR, P\_ERR, ESC\_ERR, CREDIT\_ERR на равенство «0». Бит CONNECTED должен быть равен «1». При выполнении этих условий - соединение с удаленной системой установлено.

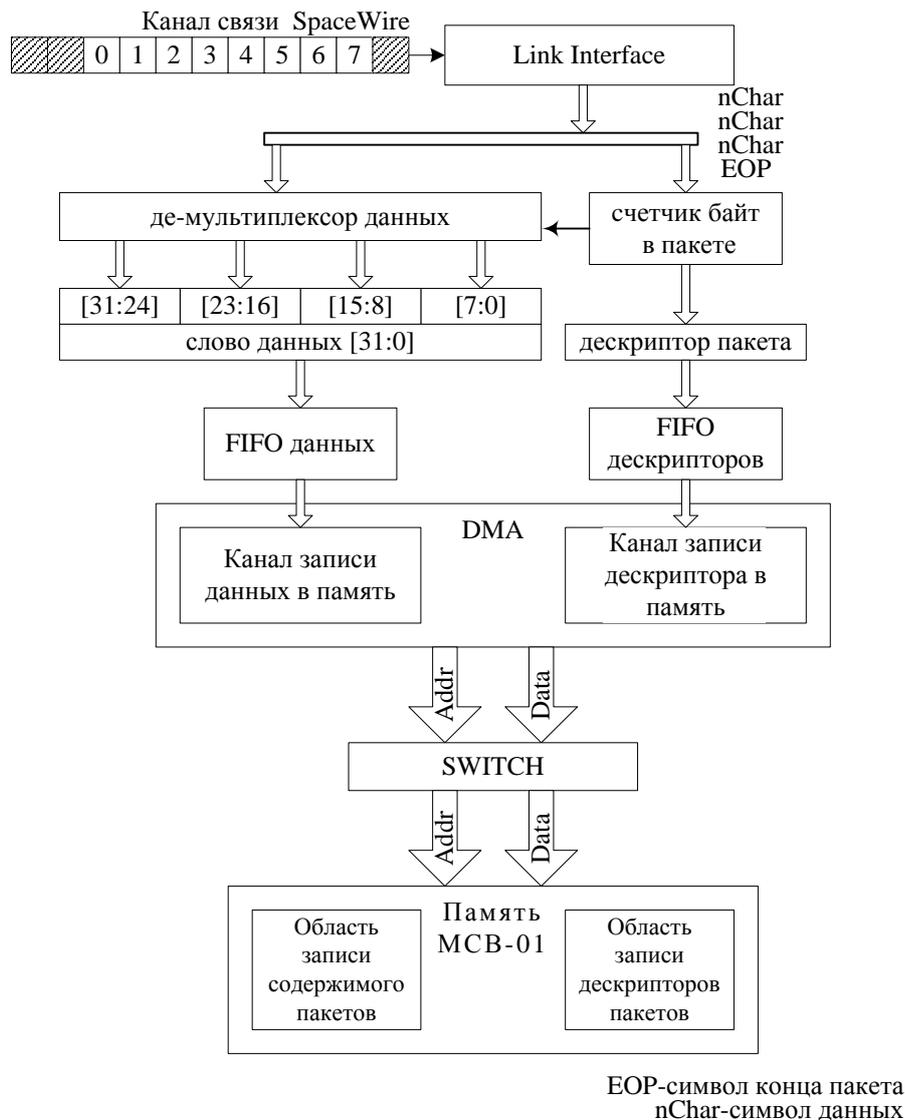
Изменение рабочей скорости передачи разрешается только после установления соединения. Рекомендуется применять адаптивный метод определения максимальной скорости передачи. После разрыва соединения, в соответствии со стандартом SpaceWire, необходимо перед повторным соединением установить скорость передачи 10 Мбит/с.

После потери соединения из-за ошибок будет выставлено прерывание INT\_ERR.

Для снижения энергопотребления, если контроллер SWIC не используется, необходимо отключить PLL\_TX.

### 5.4.2 Прием пакетов данных из канала связи SpaceWire

Схема приема пакетов данных из канала связи SpaceWire приведены на Рисунок 5.2.



**Рисунок 5.2. Схема приема пакетов данных из канала связи SpaceWire**

Данные пакетов из канала связи SpaceWire в блок Link interface поступают последовательно. Блок Link interface выделяет из последовательности приходящих символов символы данных и символы концов пакетов и передает их в блок приема. По каналу связи SpaceWire байты данных передаются младшими разрядами вперед.

Передача всех разрядов символа (9 разрядов, из них 8 используется для представления собственно байта данных, девятый бит является дополнительным и указывает, является ли этот байт символом данных nChar или символом конца пакета EOP) от блока Link interface в блок приема осуществляется в параллельном коде.

Подсчет числа символов nChar и формирование дескриптора при приеме символа конца пакета осуществляется в счетчике байт в пакете.

В блоке приема из байтов данных формируются слова разрядности 32. При формировании слов первый поступивший байт размещается в разрядах 7:0, второй – в разрядах 15:8, третий – в разрядах 23:16 и четвертый – в разрядах 31:24. Распределение символов данных по разрядам слова данных производится по счетчику байт.

Для того чтобы сократить загрузку процессора в ходе последующей обработки пакетов данных, в этом блоке выполняется выравнивание границ пакетов по границам слов и фор-

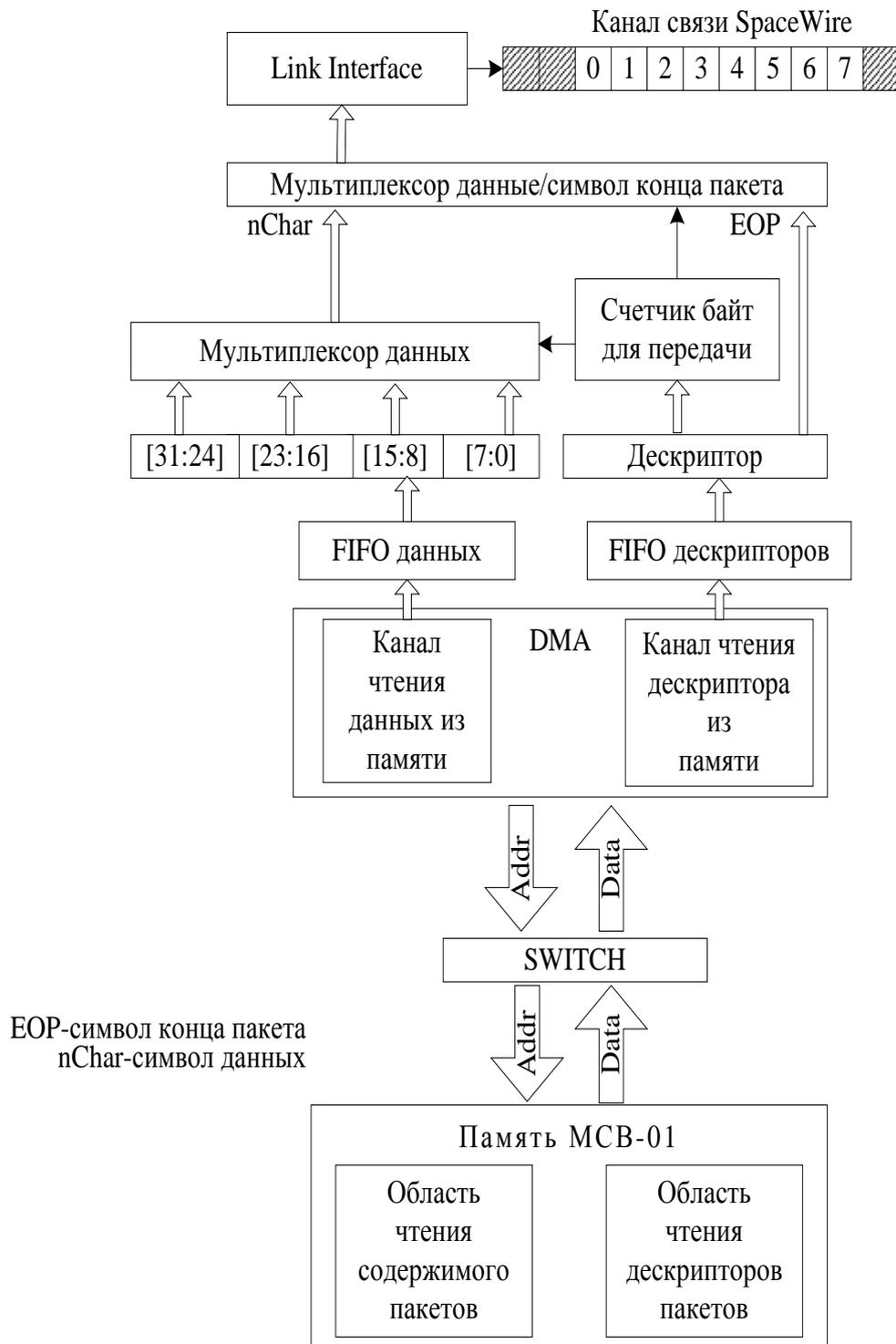
мирование дескрипторов пакетов, позволяющих процессору распознать границы отдельных пакетов.

Собственно пакеты данных и дескрипторы пакетов могут храниться в различных областях памяти. Местоположение этих областей в памяти определяется процессором при настройке каналов DMA. Дескрипторы пакетов записываются в память друг за другом и логически организованы в очередь.

Для обработки пакетов данных, принятых из канала связи SpaceWire, процессор (подключенный к MBA или Master, подключенный к PMSC) считывает первый дескриптор. По дескриптору он определяет тип конца пакета, в соответствии с этим решает, как его обрабатывать. По дескриптору он определяет действительный размер пакета и извлекает данные, относящиеся к пакету 1. Для того чтобы вычислить начальный адрес второго пакета к начальному адресу блока данных добавляется размер первого пакета и выполняется округление до границы ближайшего слова. После того, как первый пакет полностью обработан, процессор прочитывает дескриптор второго пакета. Обработка остальных пакетов выполняется аналогично. Процесс обработки очереди пакетов заканчивается, когда 31 разряд очередного дескриптора равен 0.

### 5.4.3 Передача пакетов данных в канал связи SpaceWire

Схема передачи пакетов данных из памяти микросхемы 1892ХД1Я в канал связи SpaceWire приведена на Рисунок 5.3.



**Рисунок 5.3. Схема передача пакетов данных из памяти микросхемы 1892ХД1Я в канал связи SpaceWire**

Пакеты данных загружаются из системной памяти в буфер передачи через каналы DMA чтения данных из памяти и чтения дескриптора из памяти.

Блок передачи разбивает слова на отдельные байты. При этом из последовательности байтов в соответствии с информацией, содержащейся в дескрипторе, удаляются “лишние” байты – байты, добавленные для выравнивания пакетов по границам слов, и вставляются символы концов пакетов EOP или EEP. Если в канал связи SpaceWire передаются пакеты, сгенерированные в данном узле, то предполагается, что они всегда должны заканчиваться символом EOP. Однако пакеты могут проходить через данный процессорный модуль транзитом. В этом случае они могут заканчиваться символом EEP. Коды маркеров EOP или EEP формируются контроллером аппаратно, на основании кодов дескриптора пакета на передачу (разряды 29:30 дескриптора пакета). Сами дескрипторы пакетов на передачу в сеть из основной памяти формируются программно.

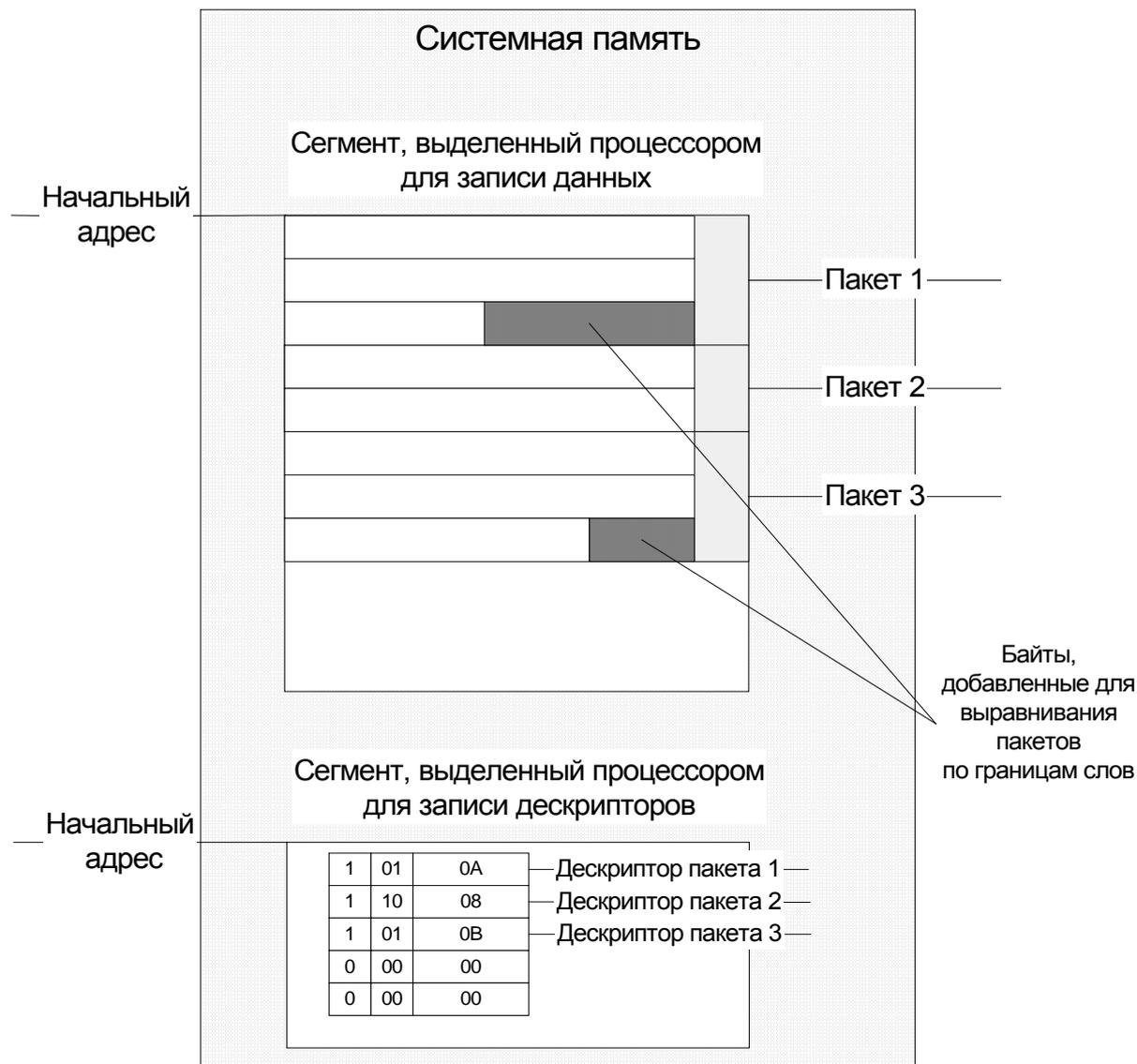
Распаковка 32-разрядного слова в последовательность из 4 байт при передаче из контроллера выполняется по правилу, согласованному с правилом упаковки байтов при приеме данных из канала в контроллер.

Блок передачи вначале передает в Link interface байт данных, находящийся в разрядах 7:0 слова, затем байт, находящийся в разрядах 15:8, затем байт, находящийся в разрядах 23:15, затем байт из разрядов 31:24 тридцатидвухразрядного слова.

Символы данных и концов пакетов передаются блоком передачи в Link interface, который преобразует полученные символы в соответствии с алгоритмом DS-кодирования и передает их в канал. Символы передаются младшими разрядами вперед.

### 5.4.4 Размещение данных и дескрипторов в памяти

На Рисунок 5.4 приведен пример размещения данных в памяти микросхемы 1892ХД1Я.



**Рисунок 5.4** Пример размещения данных и дескрипторов в памяти

Пусть в память микросхемы 1892ХД1Я из канала связи SpaceWire было записано 3 пакета. Первый пакет имеет размер 10 байт и заканчивается символом EOP. Второй пакет имеет размер 8 байт и заканчивается символом EEP. Третий пакет имеет размер 11 байт и заканчивается символом EOP. Собственно пакеты хранятся в сегменте памяти, выделенном процессором для записи данных. Первый и третий пакет дополняются двумя и одним байтом соответственно, для выравнивания по границам 32-х разрядных слов.

Дескрипторы хранятся в сегменте памяти, выделенном процессором для их записи. В дескрипторе указаны размеры пакетов в байтах – 0Ah, 08h и 0Bh соответственно. В дескрипторах хранится так же информация о типе конца пакета. В 31 разряд дескриптора записы-

вается 1, что указывает процессору на то, что дескриптор заполнен действительными данными.

Рассмотрим выравнивание пакетов данных на примере Рисунок 5.4. Если очередное слово данных сформировано не полностью (действительными данными заполнены один, два или три байта слова), а следующий символ в последовательности – символ конца пакета, то заполнение данного слова прекращается. Первый символ следующего пакета будет записан в первый байт нового слова. Действительный размер пакета в байтах записывается в дескриптор пакета. Это позволяет процессору при обработке пакета исключить из рассмотрения “лишние” байты – байты, добавленные для выравнивания пакетов по границам слов. В дескриптор заносится также информация о типе конца пакета (нормальный конец пакета – EOP, или признак завершения пакета с ошибкой – EEP).

### 5.4.5 Формат дескриптора пакета

Дескриптор пакета имеет следующую структуру:

- бит 31 – признак заполнения дескриптора действительными данными. Бит учитывается только при приёме пакетов (позволяет процессору идентифицировать конец очереди дескрипторов в памяти). В дескрипторах для передачи пакетов этот бит необходимо установить в единичное состояние. До запуска приёма, все 31-е биты дескрипторов области приёма должны быть обнулены программно; DMA не обнуляет 31-е биты не принятых дескрипторов, а только записывает ‘1’ в 31-е биты принятых дескрипторов;
- биты 30:29 – тип конца пакета: 01 – EOP, 10 – EEP;
- биты 28:25 – не используются (0000);
- биты 24:0 – размер пакета в байтах.

Слова данных из буфера приема передаются в канал DMA записи данных в память. Дескрипторы из блока приема передаются в канал DMA записи дескриптора в память. Блок DMA записывает данные и дескрипторы в системную память в соответствии с настройками, выполненными процессором.

Процессор для канала записи дескрипторов в память определяет начальный адрес блока памяти и размер блока памяти. Для записи собственно пакетов данных в память может быть задан один блок памяти (так же, как и для канала записи дескриптора в память) или последовательность блоков памяти, физически расположенных в разных местах памяти.

### 5.4.6 Маркеры времени

Маркеры времени - системная функция стандарта SpaceWire. Они предназначены для синхронизации системных часов взаимодействующих систем.

При передаче данных маркеры времени имеют наивысший приоритет. Маркер времени записывается в регистр TX\_CODE. Этот же регистр используется и для передачи в сеть кодов распределенных прерываний и roll-кодов. Запись в регистр TX\_CODE допустима только при FL\_CONTROL=0 в регистре STATUS. После записи в регистр TX\_CODE блок Link interface ожидает окончания передачи символа данных или служебного символа и

начинает передачу маркера времени. После окончания передачи маркера времени передача потока данных возобновляется.

В канале приема маркер времени выделяется из потока данных и при безошибочном приеме заносится в регистр `RX_CODE[7:0]` с выставлением соответствующего прерывания, если маркер времени является корректным. Корректным признается маркер времени на 1 больше, чем предыдущий, если предыдущий маркер времени имел значение меньше 63. Если предыдущий маркер времени имел значение 63, то следующий корректный маркер времени должен иметь значение 0. Если маркер времени не является корректным, то его значение так же заносится в соответствующие разряды регистра `RX_CODE`, однако, прерывание для процессора в данном случае не устанавливается. В начале работы устройства или после сброса маркер времени со значением 1 рассматривается как корректный.

### 5.4.7 Коды распределенных прерываний

Коды распределенных прерываний являются расширением стандарта SpaceWire. Механизм передачи кодов распределенных прерываний в сеть аналогичен механизму передачи маркеров времени.

При приеме кода распределенного прерывания из сети выполняются следующие действия. Если соответствующий коду распределенного прерывания разряд регистра `ISR` установлен в 1, то данное прерывание игнорируется (никаких действий не выполняется). Если соответствующий разряд регистра установлен в 0, то в него записывается 1 и код распределенного прерывания записывается в регистр `RX_CODE[15:8]`. В этом случае устанавливается прерывание.

### 5.4.8 Roll-коды

Roll-коды являются расширением стандарта SpaceWire. Механизм передачи roll-кодов в сеть аналогичен механизму передачи маркеров времени.

При приеме roll-кода прерывания из сети выполняются следующие действия. Если соответствующий roll-коду разряд регистра `ISR` установлен в 0, то данный код игнорируется (никаких действий не выполняется). Если соответствующий разряд регистра установлен в 1, то в него записывается 0 и код записывается в регистр `RX_CODE[23:16]`. В этом случае устанавливается прерывание.

## 5.5 Логика работы DMA SWIC

### 5.5.1 Типы каналов

Каждый контроллер SWIC имеет свой 4-канальный контроллер DMA. Выбор контроллера осуществляется по базовому адресу, приведенному в Таблица 5.19.

**Таблица 5.19 Базовые адреса контроллеров DMA SWIC**

Условное обозначение контроллера	Базовый адрес
DMA SWIC0	0x150_0000
DMA SWIC1	0x170_0000
DMA SWIC2	0x190_0000
DMA SWIC3	0x1B0_0000

Перечень каналов DMA и их взаимный приоритет приведен в Таблица 5.20.

**Таблица 5.20 Перечень каналов DMA**

Условное обозначение канала	Назначение канала	Приоритет канала DMA	Приоритет контроллера DMA
<u>Каналы DMA SWIC0</u>			
SWIC0_RX_DES C	Канал записи в память дескрипторов принимаемых пакетов	0	0
SWIC0_RX_DATA	Канал записи в память принимаемых слов данных	1	
SWIC0_TX_DES C	Канал чтения из памяти дескрипторов передаваемых пакетов	2	
SWIC0_TX_DATA	Канал чтения из памяти передаваемых слов данных	3	
<u>Каналы DMA SWIC1</u>			
SWIC1_RX_DES C	Канал записи в память дескрипторов принимаемых пакетов	0	1
SWIC1_RX_DATA	Канал записи в память принимаемых слов данных	1	
SWIC1_TX_DES C	Канал чтения из памяти дескрипторов передаваемых пакетов	2	
SWIC1_TX_DATA	Канал чтения из памяти передаваемых слов данных	3	
<u>Каналы DMA SWIC2</u>			
SWIC2_RX_DES C	Канал записи в память дескрипторов принимаемых пакетов	0	2
SWIC2_RX_DATA	Канал записи в память принимаемых слов данных	1	
SWIC2_TX_DES C	Канал чтения из памяти дескрипторов передаваемых пакетов	2	
SWIC2_TX_DATA	Канал чтения из памяти передаваемых слов данных	3	
<u>Каналы DMA SWIC3</u>			
SWIC3_RX_DES C	Канал записи в память дескрипторов принимаемых пакетов	0	3
SWIC3_RX_DATA	Канал записи в память принимаемых слов данных	1	
SWIC3_TX_DES C	Канал чтения из памяти дескрипторов передаваемых пакетов	2	
SWIC3_TX_DATA	Канал чтения из памяти передаваемых слов данных	3	

Взаимный приоритет контроллеров и каналов DMA представлен в таблице в порядке убывания: 0- высший приоритет, 3 – низший.

Канал DMA за один цикл занятия коммутатора SWITCH передает одно 32-разрядное слово данных за один такт частоты CLK. Если после передачи слова нет запросов от более приоритетных каналов DMA, то канал начинает передавать следующее слово данных.

## 5.5.2 Регистры DMA

Для управление работой канала DMA имеются следующие регистры:

- регистр управления и состояния CSR;
- индексный регистр IR;
- регистр адреса блока параметров CP;
- псевдорегистр управления битом RUN.

Все регистры канала DMA доступны по чтению и записи.

Формат регистра управления и состояния CSR приведен в Таблица 5.21.

**Таблица 5.21 Формат регистра CSR**

Номер разряда	Условное Обозначение	Назначение
31:16	WC	Число 32-разрядных слов данных, которые должен передать канал DMA (блок данных): 0x0 - 1 слово; 0x1 – 2 слова; 0x2 – 3 слова; ... 0xFFFF - 65536 слов. Содержимое этого поля уменьшается на 1 после передачи каналом DMA очередного 32-разрядного слова
15	DONE	Признак завершения передачи блока данных. Устанавливается в 1: при записи 1 в этот разряд; по завершению передачи блока данных при CHEN=0 (CHEN=1 может быть только при использовании процедуры самоинициализации). Устанавливается в 0: при записи 0 в этот разряд; при чтении содержимого этого регистра
14	END	Признак окончания передачи блока данных. Используется только при обмене цепочкой блоков данных. Устанавливается в 1: при записи 1 в этот разряд; при завершении передачи блока данных. Устанавливается в 0: при записи 0 в этот разряд; при чтении содержимого этого регистра; после выполнения процедуры самоинициализации, если IM=0

Номер разряда	Условное Обозначение	Назначение
13	IM	Маска формирования прерывания при END=1: 0 – формирование прерывания запрещено; после выполнения процедуры самоинициализации бит END сбрасывается; 1 – формирование прерывания разрешено; после выполнения процедуры самоинициализации бит END не сбрасывается. Используется только при обмене цепочкой блоков данных. Состояние этого бита определяется в процессе выполнения процедуры самоинициализации
12	CHEN	Разрешение выполнения очередной процедуры самоинициализации: 0 – выполнение очередной процедуры самоинициализации запрещено; 1 – выполнение очередной процедуры самоинициализации разрешено. Используется только при обмене цепочкой блоков данных. Состояние этого бита определяется в процессе выполнения процедуры самоинициализации
11:6	-	Не используется
5:2	WN	Размер пакета слов данных, передаваемых за 1 предоставления доступа DMA к коммутатору SWITCH: 0x0 – 1 слово; 0x1 – 2 слова; ... 0xF – 16 слов. Используется для задания темпа передачи данных по DMA
1	-	Не используется
0	RUN	Состояние работы канала DMA: 0 – состояние останова; 1 – состояние обмена данными. Устанавливается в 1 при записи 1 в этот разряд. Устанавливается в 0: при записи 0 в этот разряд; при завершении передачи блока данных. Состояние этого бита определяется в процессе выполнения процедуры самоинициализации

Канал DMA устанавливает соответствующее прерывание в регистрах QSTR и QSTR\_PCI при DONE=1 или END&IM.

Исходное состояние регистра CSR: разряды [15:0] – нули, а состояние разрядов [31:16] не определено.

32-разрядный индексный регистр содержит физический адрес первого слова блока данных. Исходное состояние не определено. Младшие два разряда IR должны быть равны нулю. После передачи каждого слова регистр IR аппаратно инкрементируется на 4.

Формат регистра адреса блока параметров CP приведен в Таблица 5.22.

**Таблица 5.22 Формат регистра CP**

Номер разряда	Условное обозначение	Назначение
31:1	CP	Адрес блока параметров DMA-передачи для выполнения процедуры самоинициализации. Адрес должен быть выровнен по границе 32-разрядного слова.
0	AUTO_INIT	При выполнении операции записи 1 в этот разряд осуществляется запуск процедуры самоинициализации. При чтении – всегда 0

При помощи 1-разрядного псевдорегистра управления битом RUN (соответствует разряду 0 шины данных) можно управлять состоянием разряда RUN регистра CSR, не изменяя содержимого других разрядов. То есть, останавливать канал DMA или вновь его запускать.

### 5.5.3 Запуск канала DMA при помощи регистра CSR

Запуск канала DMA при помощи регистра CSR используется для передачи одного блока данных.

Перед запуском канала DMA необходимо убедиться в том, что он находится в состоянии останова (RUN = 0). Затем необходимо записать:

- адрес первого слова данных в регистр IR;
- в поле WC регистра CSR число слов в блоке данных, а в разряды 15:0 – код 0x0001 (IM=0, DONE=0, CHEN=0, END=0, RUN=1).

### 5.5.4 Выполнение процедуры самоинициализации

Все каналы DMA SWIC могут выполнять процедуру самоинициализации для выполнения обмена цепочкой блоков данных. Процедура самоинициализации используется для передачи нескольких блоков данных.

Для выполнения самоинициализации в каналах имеется 32-разрядный регистр CP, в котором хранится начальный адрес блока параметров очередного DMA-обмена. Эти параметры при самоинициализации аппаратно загружаются в соответствующие регистры канала DMA. Процедура этой загрузки ничем не отличается от обычного DMA-обмена. Блок параметров размещается в памяти DPRAM.

Блоки параметров, размещаемых в памяти, имеют следующую структуру (в порядке возрастания адресов) – IR, CSR, CP:

**Таблица 5.23**

Параметр	Смещение
IR	0x00
CP	0x04
CSR	0x08

Параметры, соответствующие 16-разрядным регистрам, размещаются в младших разрядах памяти. В слове памяти, соответствующем регистру CSR, должно быть: RUN=1, DONE=0. Если необходимо продолжить цепочку команд, то необходимо указать CHEN=1.

Для запуска работы канала DMA в режиме с самоинициализацией необходимо в регистр CP записать адрес первого блока параметров DMA-передачи. При этом 0 разряд записываемых данных должен содержать 1 (признак пуска самоинициализации). В результате этого, соответствующий канал загрузит в свои регистры параметры DMA передачи и начнет обмен данными.

После окончания передачи данного блока данных устанавливается в единичное состояние бит END в регистре CSR и выдается прерывание, если бит IM = 1. После этого канал проверяет состояние бита CHEN. Если он равен 1, то будет загружен следующий блок пара-

метров DMA передачи и т.д. В противном случае цепочка DMA обменов закончится и в регистре CSR бит DONE установится в единичное состояние.

## 5.6 Примеры работы со SWIC

Для передачи пакета данных в канал связи SpaceWire необходимо:

- - установить соединение с соседним устройством, подключенным к каналу связи SpaceWire;
- - записать в память DPRAM пакет для передачи;
- - сформировать дескриптор этого пакета;
- - настроить передающий канал DMA данных;
- - настроить передающий канал DMA дескриптора;
- - запустить настроенные каналы DMA;
- - дождаться окончания работы каналов DMA.

Для приема пакета данных из канала связи SpaceWire необходимо:

- - установить соединение с соседним устройством, подключенным к каналу связи SpaceWire;
- - настроить приемный канал DMA дескрипторов;
- - настроить приемный канал DMA данных;
- - запустить настроенные каналы DMA;
- - дождаться окончания работы каналов DMA;
- - считать принятый пакет данных из DPRAM.

### 5.6.1 Процедура передачи пакета

Рассмотрим подробнее процедуру передачи пакета:

```

TX_SPEED = 0x302 //Устанавливаем скорость соединения 10Мбит/сек.
MODE_CR = LINK_START // Запускаем соединение
STATUS [7:5] = 3'B101? // Чтением регистра статуса ожидаем установ-
ления соединения
// Готовим данные в памяти
DPRAM@ 0x01000100 = 0xA0000005 // Дескриптор на передачу, 5 байт к пе-
редаче.
DPRAM@ 0x01000200 = 0xA4A3A2A1 // данные на передачу
DPRAM@ 0x01000204 = 0xA8A7A6A5 // данные на передачу
// Настраиваем канал SWIC_TX_DESC
SWICx_TX_DESC_CSR = 0x00002000 // Одно слово к передаче + установка
прерывания
SWICx_TX_DESC_IR = 0x01000100 // Адрес, где лежит дескриптор пакета
для передачи
SWICx_TX_DESC_CP = 0x0 // отключаем цепочечные передачи
// Настраиваем канал SWIC_TX_DATA
SWICx_TX_DATA_CSR = 0x00012000// Два слова к передаче
SWICx_TX_DATA_IR = 0x01000200 // Адрес, где лежит первое слово пакета
для передачи
SWICx_TX_DATA_CP = 0x0 // отключаем цепочечные передачи
// Запускаем каналы передачи
SWICx_TX_DESC_RUN = 0x1
SWICx_TX_DATA_RUN = 0x1
// Ожидание завершения работы каналов
SWICx_TX_DESC_RUN[0] & SWICx_TX_DATA_RUN[0] = 0? // Чтением регистров
статуса ожидаем завершения работы каналов DMA.

```

## 5.6.2 Процедура приема пакета

Рассмотрим подробнее процедуру приема пакета (получаем тот пакет, что передавали ранее):

```

// Предположим, что соединение уже установлено
// Настраиваем канал SWIC_RX_DESC
SWICx_RX_DESC_CSR = 0x00002000// Одно слово к передаче (из SWIC в па-
мять MCB) + установка прерывания
SWICx_RX_DESC_IR = 0x01000300 //Адрес, где сохранится дескриптор при-
нятого пакета
SWICx_RX_DESC_CP = 0x0 // отключаем цепочечные передачи
// Настраиваем канал SWIC_RX_DATA
SWICx_RX_DATA_CSR = 0x00012000// Два слова к передаче (из SWIC в па-
мять MCB) + установка прерывания
SWICx_RX_DESC_IR = 0x01000400 // Адрес, где сохранится первое слово
принятого пакета
SWICx_RX_DESC_CP = 0x0 // отключаем цепочечные передачи
// Запускаем каналы передачи
SWICx_RX_DESC_RUN = 0x1
SWICx_RX_DATA_RUN = 0x1
// Ожидание завершения работы каналов
SWICx_RX_DESC_RUN[0] & SWICx_RX_DATA_RUN[0] = 0? // Чтением регистров
статуса ожидаем завершения работы каналов DMA.
// Разбираем принятые данные
Читаем DPRAM@ 0x01000300 = 0x2000_0005 ? // Принят пакет без ошибки,
длиной 5 байт.
Читаем DPRAM@ 0x01000400 = 0xA4A3_A2A1 ? // Первые 4 байта пакета. Чи-
таем DPRAM@ 0x01000404 = 0x0000_00A5 ? // Пятый байт пакета. Старшие
три байта пусты.

```

## 6. РЕКОМЕНДАЦИИ ПО ПРИМЕНЕНИЮ

### 6.1 Рекомендации по подключению failsafe резисторов

Приемники LVDS микросхемы 1892ХД1Я не обладают свойствами failsafe. То есть, на их выходе не обеспечивается определенное состояние, при отсутствии входного сигнала. Это может приводить к ложному соединению при обрыве входного кабеля или при его отсутствии.

Для обеспечения функции failsafe необходимо через резисторы 1,3 кОм $\pm$ 5% подключить:

- входы DIN<sub>p</sub>, SIN<sub>p</sub> к шине GND микросхемы;
- входы DIN<sub>n</sub>, SIN<sub>n</sub> к напряжению электропитания PVDD микросхемы.

## 7. ЭЛЕКТРИЧЕСКИЕ ПАРАМЕТРЫ

### 7.1 Электропитание

Номинальное значение напряжения электропитания микросхемы:

$U_{CC1}=3,3$  В (периферия);

$U_{CC2}=2,5$  В (ядро).

Допустимые отклонения напряжения электропитания микросхемы от номинального значения - не более  $\pm 5\%$ .

При включении на микросхему сначала необходимо подать напряжение электропитания ядра (CVDD)  $U_{CC2}$ , а затем напряжение электропитания периферии (PVDD)  $U_{CC1}$ . Задержка между подачей напряжений электропитания должна быть не более 100 мс. Входные сигналы подаются после подачи напряжений электропитания или одновременно с напряжением электропитания периферии  $U_{CC1}$ . Фронт нарастания напряжений электропитания должен быть не более 5 мс;

При выключении микросхемы необходимо сначала снять входные сигналы, затем напряжение электропитания периферии  $U_{CC1}$ , затем, с задержкой не более 100 мс, напряжение электропитания ядра  $U_{CC2}$ .

Для фильтрации напряжений электропитания микросхемы, необходимо подключить к каждому источнику ( $U_{CC1}$  и  $U_{CC2}$ ) не менее шести высокочастотных конденсаторов номиналом 0,1 мкФ типа СС 0603 Y5V 0,1  $\mu$ F Z 25V. Конденсаторы необходимо разместить по возможности равномерно по контуру корпуса микросхемы между выводами PVDD и GND, а так же CVDD и GND. При этом расстояние между контактами микросхемы и площадками подсоединения конденсаторов должно быть не более 3 мм.

### 7.2 Электрические параметры при приемке и поставке

Электрические параметры микросхемы приведены в Таблица 6.1.

**Таблица 6.1. Электрические параметры микросхемы**

Наименование параметров, единица измерения, режим измерения	Буквенное обозначение	Норма	
		не менее	не более
Ток потребления статический по цепи PVDD, мА при $U_{CC1}=3,47$ В, $U_{CC2}=2,63$ В, ХТI=0	$I_{CC1}$	-	10
Ток потребления статический по цепи CVDD, мА при $U_{CC1}=3,47$ В, $U_{CC2}=2,63$ В, ХТI=0	$I_{CC2}$	-	15
Ток потребления динамический по цепи CVDD, мА, при $U_{CC1}= 3,47$ В, $U_{CC2}= 2,63$ В и рабочей ча- стоте 100 МГц	$I_{OCC2}$	-	200

Наименование параметров, единица измерения, режим измерения	Буквенное обозначение	Норма	
		не менее	не более
Ток утечки высокого и низкого уровня на входе, мкА при $U_{CC1}=3,47$ В и $U_{CC2}=2,63$ В	$I_{IL}$	-	2
Выходное напряжение низкого уровня, В при $I_{OL}=4$ мА, $U_{CC1}=3,47$ В	$U_{OL}$	-	0,4
Выходное напряжение высокого уровня, В при $I_{OH}=-2,8$ мА, $U_{CC1}=3,13$ В	$U_{OH}$	2,4	
Входная емкость, пФ	$C_I$	-	10
Емкость входа/выхода, пФ	$C_{I/O}$	-	10
Выходная емкость, пФ	$C_O$	-	15
Выходное дифференциальное напряжение по LVDS выходам, мВ	$U_{OD}$	300	400
Входное дифференциальное сопротивление по LVDS входам, Ом	$R_{ID}$	90	120

### 7.3 Предельно-допустимые и предельные электрические режимы эксплуатации

Значения предельно-допустимых и предельных электрических режимов эксплуатации микросхемы приведены в Таблица 6.2.

**Таблица 6.2. Значения предельно-допустимых и предельных электрических режимов эксплуатации**

Наименование параметра	Буквенное обозначение	Норма			
		Предельно допустимый режим		Предельный Режим	
		не менее	не более	не менее	не более
Напряжение питания периферии, В	$U_{CC1}$	3,13	3,47	-	3,9
Напряжение питания ядра, В	$U_{CC2}$	2,37	2,63	-	3,0
Входное напряжение высокого уровня, В	$U_{IH}$	2,0	$U_{CC1}+0,2$	-	$U_{CC1}+0,3$
Входное напряжение низкого уровня, В	$U_{IL}$	0,0	0,7	-0,3	-
Напряжение, прикладываемое к выходу микросхемы в состоянии «выключено», В	$U_{OZ}$	0,0	$U_{CC1}+0,1$	-0,3	$U_{CC1}+0,3$
Емкость нагрузки каждого выхода, пФ	$C_L$	-	30	-	50

## 8. ОПИСАНИЕ ВНЕШНИХ ВЫВОДОВ МИКРОСХЕМЫ

### 8.1 Назначение выводов микросхемы 1892ХД1Я

Микросхема 1892ХД1Я имеет следующие группы выводов:

- шина обмена данными с внешним микропроцессором (мастером);
- шина PCI;
- 4 порта канала связи SpaceWire;
- управление;
- электропитание.

Описание выводов приведено в таблицах 7.1-7.5.

Неиспользуемые выводы типа «I» необходимо подключить к земле.

Неиспользуемые выводы типа «O» необходимо оставить неподключенными.

Каждый неиспользуемый вывод типа «IO» необходимо подключить к земле через резистор 10 кОм.

При описании выводов используются обозначения:

- I – вход;
- O – выход;
- IO – двунаправленный вход/выход с третьим состоянием;
- OT – выход с третьим состоянием. При низком уровне nRST находится в пассивном состоянии;
- OD – выход с открытым коллектором, возможно объединение по схеме проводного ИЛИ;
- префикс “n” (например nWE) означает, что активный уровень сигнала - низкий. Иначе, у сигнала активный высокий уровень

**Таблица 8.1 Шина обмена данными с внешним микропроцессором**

Название вывода	Количество	Тип	Назначение
A[24:0]	25	I	Шина адреса. Содержит адрес 32-разрядного слова в адресном пространстве микросхемы 1892ХД1Я. При обмене данными через адаптер МВА внутренний адрес микросхемы 1892ХД1Я формируется посредством добавления к разрядам A[22:0] двух нулей справа. Разряды A[24:23] используются для сравнения с номером N[1:0]. Следует иметь в виду, что по шине A[24:0] передается адрес 32-разрядного слова. Поэтому при сопряжении микросхемы 1892ХД1Я с микропроцессорами серии «Мультикор» эту шину необходимо подключить к шине адреса микропроцессора начиная со 2 разряда (нулевой разряд шины A[24:0] подключить ко 2 разряду шины адреса микропроцессора и т.д.)
D[31:0]	32	Ю	Шина данных.
nWE[3:0]	4	I	Сигналы записи байтов.
nRD	1	I	Сигнал чтения.
nCS	1	I	Разрешение выборки.
nACK	1	OD	Сигнал готовности данных
nINT	1	O	Запрос прерывания: все незамаскированные прерывания регистра QSTR объединенные по ИЛИ
Всего 65 выводов			

Таблица 8.2 Управление

Название вывода	Количество	Тип	Назначение
PLL_EN	1	I	Разрешение работы PLLx20 (см. Рисунок 2.4): 0 – PLLx20 отключена. Частота MBA_CLK равна частоте ХТИ. Тестовый режим работы микросхемы 1892ХД1Я. 1 - PLLx20 включена. Частота MBA_CLK равна частоте ХТИ умноженной 20. Штатный режим работы микросхемы 1892ХД1Я. В штатном режиме работы данный вывод должен быть соединен с выводом nRST.
ХТИ	1	I	Системная тактовая частота. Тактовая частота 10 МГц +-1% Стабильность частоты – не хуже +50 ppm, скважность – от 40 до 60%, джиттер – не более 1 %
nRST	1	I	Сигнал установки исходного состояния. Во время действия сигнала nRST все узлы микросхема находится в исходном (неактивном) состоянии, выходы - в неактивном состоянии, входы-выходы являются входами При включении электропитания микросхемы сигнал nRST должен иметь низкий уровень и переключаться на высокий уровень через время не менее 1 мс после установки стабильного электропитания и стабильной тактовой частоты на входе ХТИ. Если необходимо установить работающую микросхему в исходное состояние, то для этого на нее необходимо подать асинхронный сигнал nRST длительностью не менее 10 тактов частоты на входе ХТИ. Фронт и спад сигнала nRST должен быть не более 100 нс.
N[1:0]	2	I	Номер микросхемы. Используется при подключении нескольких микросхем 1892ХД1Я
IN, nOUT	2	I, O	Выводы инвертора. Используются при необходимости
IN00, IN01, OUT0	3	I, O	Выводы первой схемы ИЛИ. Используются при необходимости.
IN10, IN11, OUT1	3	I, O	Выводы второй схемы ИЛИ. Используются при необходимости.
Всего 13 выводов			

Таблица 8.3 Порты каналов связи SpaceWire

Название вывода	Количество	Тип	Назначение
DINp[3:0]	4	I	Вход данных положительный
DINn[3:0]	4	I	Вход данных отрицательный
SINp[3:0]	4	I	Вход строба положительный
SINn[3:0]	4	I	Вход строба отрицательный
DOUp[3:0]	4	O	Выход данных положительный
DOUn[3:0]	4	O	Выход данных отрицательный
SOUUp[3:0]	4	O	Выход строба положительный
SOUUn[3:0]	4	O	Выход строба отрицательный
Всего 32 вывода			

Таблица 8.4 Шина PCI

Наименование сигнала	Количество	Тип	Назначение
AD[31:0]	32	IO	Адрес/Данные
nCBE[3:0]	4	IO	Команда/ выбор байта
nFRAME	1	IO	Признак выполнения операции передачи данных
nIRDY	1	IO	Готовность устройства master
nTRDY	1	IO	Готовность устройства target
nSTOP	1	IO	Признак остановки передачи данных
PAR	1	IO	Дополнение до чётности количества единиц на шинах AD и nCBE
nPERR	1	IO	Ошибка чётности
nDEVSEL	1	IO	Подтверждения выборки
IDSEL	1	I	Выборка при доступе к конфигурационным регистрам
nREQ	1	OT	Запрос захвата шины
nGNT	1	I	Разрешение захвата шины
nINTA	1	OD	Прерывание. Устанавливается низкий уровень если биты регистра Status/Command находятся в состоянии: Interrupt Disable = 0, Interrupt Status = 1
PCLK	1	I	Тактовая частота работы шины PCI.
nREQB[4:0]	5	I	Запрос на использование шины PCI.
nGNTB[4:0]	5	O	Разрешение использования шины PCI.
всего 58 выводов			

Таблица 8.5 Электропитание

Название вывода	Количество	Назначение
CVDD	21	Напряжение электропитания ядра
PVDD	31	Напряжение электропитания входных и выходных драйверов
GND	84	Земля ядра
Всего 136 выводов		

Нумерация выводов микросхемы 1892ХД1Я в корпусе HSBGA-416 приведена в Таблица 7.6.

**Таблица 8.6 Нумерация выводов микросхемы 1892ХД1Я в корпусе HSBGA-416**

№ вывода корпуса	Условное обозначение	№ вывода корпуса	Условное обозначение
H1	AD<24>	AC8	D<13>
K1	AD<25>	AF11	D<14>
J1	AD<26>	AE11	D<15>
C1	AD<27>	AC11	D<16>
B1	AD<28>	AD11	D<17>
D1	AD<29>	AE10	D<18>
E1	AD<30>	AF10	D<19>
L1	AD<31>	AD10	D<20>
H3	DINP<0>	AC10	D<21>
F4	DINN<0>	AF15	D<22>
E4	SINP<0>	AE15	D<23>
G4	SINN<0>	AD15	D<24>
H4	DOUDP<0>	AC15	D<25>
L2	DOUDN<0>	AF16	D<26>
M2	SOUTP<0>	AE16	D<27>
P2	SOUTN<0>	AD16	D<28>
N2	DINP<1>	AC16	D<29>
T2	DINN<1>	AF17	D<30>
R2	SINP<1>	AE17	D<31>
U2	SINN<1>	Y23	A<24>
V2	DOUDP<1>	AB26	A<23>
N4	DOUDN<1>	Y24	A<22>
P4	SOUTP<1>	AA25	A<21>
R4	SOUTN<1>	AA26	A<20>
T4	XTI	W23	A<19>
U4	nACK	W24	A<18>
V4	nRST	Y25	A<17>
L3	PLL_EN	Y26	A<16>
M3	DINP<2>	V23	A<15>
P3	DINN<2>	V24	A<14>
N3	SINP<2>	W26	A<13>
K4	SINN<2>	U23	A<12>
J4	DOUDP<2>	U24	A<11>
L4	DOUDN<2>	V25	A<10>
M4	SOUTP<2>	V26	A<9>
T1	SOUTN<2>	T23	A<8>
U1	DINP<3>	T24	A<7>
W1	DINN<3>	U25	A<6>
V1	SINP<3>	U26	A<5>
AA1	SINN<3>	R23	A<4>
Y1	DOUDP<3>	R24	A<3>
AB1	DOUDN<3>	T25	A<2>
AC1	SOUTP<3>	T26	A<1>
R3	SOUTN<3>	P23	A<0>
AC7	D<0>	P24	NRD
AD7	D<1>	R25	NCS
AE6	D<2>	R26	IN00
AF6	D<3>	N23	IN01
AD6	D<4>	N24	OUT0
AC6	D<5>	P25	IN10

№ вывода корпуса	Условное обозначение	№ вывода корпуса	Условное обозначение
AF9	D<6>	P26	IN11
AE9	D<7>		
AC9	D<8>		
AD9	D<9>		
AE8	D<10>		
AF8	D<11>		
AD8	D<12>		

Продолжение Таблица 8.6

№ вывода корпуса	Условное обозначение	№ вывода корпуса	Условное обозначение
N25	OUT1	D7	AD<21>
M25	NWE<0>	C6	AD<22>
M24	NWE<1>	B5	AD<23>
M23	NWE<2>	A1	GND
L26	NWE<3>	A12	GND
L25	NINT	A2	GND
L24	N<1>	AA23	GND
L23	N<0>	AB23	GND
K26	IN	AC14	GND
K25	NOUT	AC23	GND
D19	NREQB<0>	AC24	GND
C19	NREQB<1>	AD14	GND
B19	NREQB<2>	AD23	GND
A19	NREQB<3>	AD24	GND
D18	NREQB<4>	AD25	GND
C18	NGNTB<4>	AE14	GND
B18	NGNTB<3>	AE24	GND
A18	NGNTB<2>	AE25	GND
D17	NGNTB<1>	AE26	GND
C17	NGNTB<0>	AF14	GND
B17	NREQ	AF26	GND
A17	NGNT	AF3	GND
D16	NINTA	B12	GND
C16	IDSEL	B2	GND
B16	NDEVSEL	B3	GND
A16	NPERR	C12	GND
D15	PAR	C3	GND
C15	NSTOP	C4	GND
B15	NTRDY	D12	GND
A15	NIRDY	D4	GND
D14	NFRAME	D5	GND
C14	NCBE<3>	F1	GND
B14	NCBE<2>	K10	GND
A14	NCBE<1>	K11	GND
A10	NCBE<0>	K16	GND
B10	PCLK	K17	GND
D10	AD<0>	K3	GND
C10	AD<1>	L10	GND
B11	AD<2>	L11	GND
A11	AD<3>	L12	GND
C11	AD<4>	L13	GND
D11	AD<5>	L14	GND
A8	AD<6>	L15	GND
B8	AD<7>	L16	GND
D8	AD<8>	L17	GND
C8	AD<9>	M11	GND

№ вывода корпуса	Условное обозначение	№ вывода корпуса	Условное обозначение
B9	AD<10>	M12	GND
A9	AD<11>	M13	GND
C9	AD<12>	M14	GND
D9	AD<13>	M15	GND
A5	AD<14>	M16	GND
A4	AD<15>	N11	GND
B6	AD<16>	N12	GND
A6	AD<17>	N13	GND
B7	AD<18>	N14	GND
A7	AD<19>	N15	GND
C7	AD<20>	N16	GND

Продолжение Таблица 8.6

№ вывода корпуса	Условное обозначение	№ вывода корпуса	Условное обозначение
N26	GND	A3	PVDD
P11	GND	AC12	PVDD
P12	GND	AC13	PVDD
P13	GND	AC4	PVDD
P14	GND	AC5	PVDD
P15	GND	AD12	PVDD
P16	GND	AD13	PVDD
R1	GND	AD3	PVDD
R11	GND	AD4	PVDD
R12	GND	AD5	PVDD
R13	GND	AE12	PVDD
R14	GND	AE13	PVDD
R15	GND	AE2	PVDD
R16	GND	AE3	PVDD
T10	GND	AF1	PVDD
T11	GND	AF12	PVDD
T12	GND	AF13	PVDD
T13	GND	AF2	PVDD
T14	GND	AF25	PVDD
T15	GND	B4	PVDD
T16	GND	C5	PVDD
T17	GND	D6	PVDD
U10	GND	K12	PVDD
U11	GND	K13	PVDD
U16	GND	M10	PVDD
U17	GND	M26	PVDD
W3	GND	N10	PVDD
AC20	GND	P17	PVDD
AE21	GND	R17	PVDD
AC21	GND	U14	PVDD
		U15	PVDD
		A13	CVDD
		A25	CVDD
		A26	CVDD
		B13	CVDD
		B24	CVDD
		B25	CVDD
		C13	CVDD
		C23	CVDD
		C24	CVDD
		D13	CVDD
		D22	CVDD
		D23	CVDD
		K14	CVDD
		K15	CVDD
		M17	CVDD

№ вывода корпуса	Условное обозначение	№ вывода корпуса	Условное обозначение
		N17	CVDD
		P10	CVDD
		R10	CVDD
		U12	CVDD
		U13	CVDD
		AF22	CVDD

## 9. ИСТОРИЯ ИЗМЕНЕНИЙ

### 9.1 17 июля 2013 г

- Дополнен раздел 3 в части обмена данными по МВА с использованием сигнала nACK.
- Дополнен раздел 4 в части выполнения на шине PCI процедуры RETRY в режиме Master.
- Дополнен раздел 7 в части введения сигнала nACK.

### 9.2 22 августа 2013 г

- Дополнен раздел 5 в части описания регистров контроллера SWIC.

### 9.3 16 сентября 2013 г

- Скорректированы пункты 5.4.5 и 5.6 в части необходимости устанавливать в единичное состояние 31 бит дескриптора для передачи данных в SWIC.

### 9.4 24 сентября 2013 г

- Уточнена табл. 7.5 в части числа выводов CVDD, GND.

### 9.5 01 ноября 2013 г

- Уточнены пункты 1.2, 5.3.5 в скорости передачи данных.

### 9.6 31 марта 2014 г

- Уточнено описание регистра MODE\_CR (таблица 5.6): удалены разряды DSM\_RST, TX\_SINGLE, RX\_SINGLE.
- Уточнен пункт 5.4.1 в части установки необходимых разрядов в регистре MODE\_CR при установке соединения по каналу SpaceWire.
- Уточнен формат регистра CP (таблица 5.22).
- Исправлена опечатка в названии поля WC регистра CSR в разделе 5.5.3.
- Скорректировано описание процедуры самоинициализации DMA в разделе 5.5.4.
- Скорректирован пример работы с интерфейсом SpaceWire в разделе 5.6 (исправлено значение регистра TX\_SPEED, удалены упоминания регистров OR).
- Термин DS-макроячейка в тексте изменен на «Link-интерфейс» (Link interface) в соответствии со стандартом SpaceWire.

## 9.7 11 апреля 2014 г

- Уточнены табл. 4.10, 5.1, 5.4, 5.6.

## 9.8 23 декабря 2014 г

- Скорректирован пример работы с интерфейсом SpaceWire в разделе 5.6 (исправлены значения полей WC регистров CSR DMA).

## 9.9 20 ноября 2015 г

- Скорректирована Таблица 8.2, поле PLL\_EN.
- Скорректирован пункт 5.6: названия регистров приведены в соответствие с Таблица 2.3.
- Обновлено оформление документа в соответствии с новыми требованиями.

## 9.10 13 ноября 2017 г

- Изменен пункт 7.1 в части подачи и снятия напряжения электропитания.

## 9.11 15 декабря 2017 г

- Предложение пункта 7.1 «Задержка между подачей напряжений электропитания должна быть не более 10 мс.» заменено на «Задержка между подачей напряжений электропитания должна быть не более 100 мс.»
- Предложение пункта 7.1 «При выключении микросхемы необходимо сначала снять входные сигналы, затем напряжение электропитания периферии  $U_{CC1}$ , затем, с задержкой не более 10 мс, напряжение электропитания ядра  $U_{CC2}$ .» заменено на «При выключении микросхемы необходимо сначала снять входные сигналы, затем напряжение электропитания периферии  $U_{CC1}$ , затем, с задержкой не более 100 мс, напряжение электропитания ядра  $U_{CC2}$ .»
- Скорректирована Таблица 8.1 в части описания назначения выводов XTI, nRST.
- Добавлена глава 6 РЕКОМЕНДАЦИИ ПО ПРИМЕНЕНИЮ.

## 9.12 1 февраля 2018 г

- Таблица 8.2. Удалено предложение «При этом, если к MPORT подключена память типа SDRAM, то до подачи сигнала nRST все операции обмена данными с SDRAM должны быть закончены.» - Так как микросхема не рассчитана на подключение к ней SDRAM.
- В той же таблице скорректировано назначение вывода XTI – тактовая частота изменена на 10 МГц  $\pm 1\%$ .