

**МИКРОСХЕМА ИНТЕГРАЛЬНАЯ 1892ХД2Я
(МСК-01)**

РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ

Содержание

СОДЕРЖАНИЕ	2
1. ВВЕДЕНИЕ	4
1.1 НАЗНАЧЕНИЕ	4
1.2 ОСНОВНЫЕ ХАРАКТЕРИСТИКИ	5
1.3 СТРУКТУРНАЯ СХЕМА	6
2. ПРОГРАММНАЯ МОДЕЛЬ	9
2.1 ОБЩИЕ ПОЛОЖЕНИЯ	9
2.2 РАСПРЕДЕЛЕНИЕ АДРЕСНОГО ПРОСТРАНСТВА	9
2.3 ПЕРЕЧЕНЬ РЕГИСТРОВ ПОРТОВ SPACEWIRE	9
2.4 ОПИСАНИЕ РЕГИСТРОВ ПОРТОВ SPACEWIRE	12
2.4.1 Регистр статуса – <i>Status</i>	12
2.4.2 Регистр режима работы – <i>MODE_CR</i>	13
2.4.3 Регистр коэффициента скорости передачи – <i>TX_SPEED</i>	13
2.4.4 Регистр коэффициента скорости приема – <i>RX_SPEED</i>	14
2.5 ПЕРЕЧЕНЬ РЕГИСТРОВ УПРАВЛЕНИЯ	14
2.6 ОПИСАНИЕ РЕГИСТРОВ УПРАВЛЕНИЯ	15
2.6.1 Регистр адаптивной групповой маршрутизации – <i>ADG_ROUT</i>	15
2.6.2 Регистр идентификатора MCK-01 – <i>ID_SWITCH</i>	16
2.6.3 Регистр режима работы MCK-01 – <i>SWITCH_CONTR</i>	17
2.6.4 Регистр идентификатора протокола – <i>ID_PROT</i>	17
2.6.5 Регистр идентификации сетевых линков – <i>ID_NET</i>	17
2.6.6 Регистр выходного управляющего кода – <i>CONTROL_OUT</i>	18
2.6.7 Регистр текущего системного времени – <i>CUR_TIME</i>	18
2.6.8 Регистр <i>ISR_H, L</i>	18
2.6.9 Регистр маски распределенных прерываний – <i>Int_H, L_mask</i>	18
2.6.10 Регистр маски poll кодов – <i>Poll_H, L_mask</i>	18
2.6.11 Регистр флагов установки соединения – <i>CUR_CONNECTED</i>	18
2.6.12 Регистр флагов ошибок – <i>CUR_ERRORED</i>	19
2.6.13 Регистр состояния MCK-01 – <i>SWITCH_STATE</i>	19
2.7 РЕГИСТРЫ DMA	19
2.8 ФОРМАТ ТАБЛИЦЫ МАРШРУТИЗАЦИИ	20
2.9 ОПИСАНИЕ ПРОЦЕССА ОБРАБОТКИ УПРАВЛЯЮЩИХ КОДОВ ВРЕМЕНИ В MCK-01	21
2.10 ОПИСАНИЕ ПРОЦЕССА ОБРАБОТКИ КОДОВ РАСПРЕДЕЛЕННЫХ ПРЕРЫВАНИЙ И POLL КОДОВ В MCK-0122	23
2.11 ОПИСАНИЕ ПРОЦЕССА ОБРАБОТКИ ПАКЕТОВ ДАННЫХ В MCK-01	23
2.12 ОПИСАНИЕ ЛОГИКИ РАБОТЫ ПРЕРЫВАНИЙ В MCK-01	24
3. РЕКОМЕНДАЦИИ ПО ПРОГРАММИРОВАНИЮ	27
4. ФУНКЦИОНАЛЬНОЕ ОПИСАНИЕ	29
4.1 ПОРТ SPACEWIRE	29
4.2 БЛОК РЕГИСТРОВ	30
4.3 ТАБЛИЦА МАРШРУТИЗАЦИИ	30
4.4 НЕБЛОКИРУЮЩИЙ КРОСС-КОММУТАТОР	30
4.4.1 Коммутационная матрица	32
4.4.2 Контроллер арбитража и коммутации	32
4.5 КОНТРОЛЛЕР РАСПРЕДЕЛЕНИЯ КОДОВ ВРЕМЕНИ	35
4.5.1 Компонент буферизации	36
4.5.2 Компонент определения текущего времени	36
4.6 КОНТРОЛЛЕР РАСПРЕДЕЛЕННЫХ ПРЕРЫВАНИЙ	37
4.6.1 Компонент буферизации	38
4.6.2 Компонент приема распределенных прерываний	38
4.6.3 FIFO распределенных прерываний и poll кодов	38
4.6.4 Компонент передачи распределенных прерываний	39
4.7 КОМПОНЕНТ АРБИТРАЖА УПРАВЛЯЮЩИХ КОДОВ	39
4.8 КОМПОНЕНТ ВЫБОРКИ АКТИВНОГО КАНАЛА В ГРУППЕ	39
4.9 ОЗУ ПАКЕТОВ	39
4.10 БЛОК DMA КОНФИГУРАЦИОННОГО ПОРТА	39

4.11	Блок CSR.....	40
4.12	МВА (КОНТРОЛЛЕР МPORT).....	40
4.13	Блок КОММУНИКАЦИОННОЙ СИСТЕМЫ АНВ.....	41
4.14	Мост АНВ/АНВ.....	42
5.	ИНФОРМАЦИЯ О ПРИМЕНЕНИИ.....	43
6.	ЭЛЕКТРИЧЕСКИЕ ПАРАМЕТРЫ.....	46
6.1	ЭЛЕКТРОПИТАНИЕ.....	46
6.2	ЭЛЕКТРИЧЕСКИЕ ПАРАМЕТРЫ.....	46
6.3	ПРЕДЕЛЬНО-ДОПУСТИМЫЕ И ПРЕДЕЛЬНЫЕ ЭЛЕКТРИЧЕСКИЕ РЕЖИМЫ ЭКСПЛУАТАЦИИ.....	47
7.	ОПИСАНИЕ ВНЕШНИХ ВЫВОДОВ.....	48

1. ВВЕДЕНИЕ

В данном документе представлено техническое описание микросхемы 16-канального маршрутизирующего коммутатора SpaceWire MCK-01 (SpWitch-16 –SpaceWire Routing Switch). MCK-01 коммутирует 16 высокоскоростных последовательных каналов (линков), обеспечивая прием/передачу данных в соответствии со стандартом SpaceWire (далее по тексту – каналы-линки SpaceWire). Он предназначен для применения в качестве коммуникационного компонента отечественной электронной элементной базы типа “система-на-кристалле”. Приведены технические характеристики коммутатора, описаны структура и функциональный состав, приведены указания по применению, программированию и тестированию.

1.1 Назначение

Микросхема MCK-01 может быть использована для обеспечения аппаратной поддержки функций внутрисистемных коммуникаций с использованием протокола SpaceWire. Маршрутизирующий коммутатор MCK-01 обеспечивает дуплексный прием-передачу последовательных данных по 16 каналам в соответствии со стандартом SpaceWire.

MCK-01 реализует функции коммутатора для этих 16 каналов SpaceWire, а также для внутреннего конфигурационного порта. Стандарт SpaceWire [Space engineering. SpaceWire – Links, nodes, routers and networks. ECSS-E-50-12A 24 January 2003/ ECSS Secretariat ESA-ESTEC. Requirements & Standards Division Noordwijk, The Netherlands.] разработан Европейским космическим агентством (European Space Agency) для передачи данных с использованием высокоскоростных (2– 400 Мбит/с) последовательных дуплексных каналов, отвечающих требованиям повышенной надёжности и другим специальным требованиям.

MCK-01 предназначен для построения масштабируемых коммуникационных структур (сетей SpaceWire) с высокой пропускной способностью на базе стека протоколов SpaceWire для распределенных вычислительных и управляющих комплексов, параллельных систем обработки сигналов и данных.

MCK-01 реализует маршрутизацию типа «червячный ход» с использованием всех методов адресации, определенных стандартом SpaceWire (адресация пути, логическая / регионально-логическая адресация) и коммутацию пакетов по стандарту SpaceWire с использованием метода коммутации "на лету", а также коммутацию с буферизацией. На основе адаптивной групповой маршрутизации MCK-01 обеспечивает программируемое распределение информационных потоков между терминальными (процессорными) модулями и их динамическую реконфигурацию в процессе передачи между модулями коммуникационной сети, а также возможность построения отказоустойчивых конфигураций коммуникационной сети.

MCK-01 поддерживает организацию распределенной и параллельной обработки информации и управления в реальном масштабе времени: организацию системы единого времени и распределенных прерываний для терминальных модулей в распределенном комплексе, а также обеспечивает минимальные накладные расходы на передачу полезной информации. MCK-01, разработанный в соответствии с многоуровневым стеком протоколов SpaceWire, позволяет обеспечить взаимодействие модулей распределенных вычислительных комплексов и параллельных ВС в широком диапазоне возможностей, от передачи разнородного потока коротких пакетов с использованием технологии виртуальных каналов до непрерывного однородного потока данных, например, от датчиков к DSP.

MCK-01 обеспечивает коммутацию «на лету» между 16 каналами, соответствующими стандарту SpaceWire. Передача пакетов осуществляется со скоростью 2–400 Мбит/с по

каждому из каналов SpaceWire в каждом направлении. Обеспечивается автоматическая адаптация приемника к скорости передатчика каждого из каналов и может быть реализована индивидуальная настройка скоростей по каждому из каналов. На физическом уровне канала-линка применяются LVDS-сигналы (стандарт ANSI/TIA/EIA-644). Диаметр коммуникационной системы с использованием МСК-01 – от 20 метров (при использовании 1 коммутатора) до 100 м и более, при построении распределенных коммуникационных структур с использованием СБИС МСК-01 в качестве сетевых узлов.

1.2 Основные характеристики

- Реализация МСК-01 охватывает уровни стека протоколов SpaceWire: сигнальный, символьный, обмена, пакетов и сетевой уровни.
- МСК-01 обеспечивает объединение шестнадцати дуплексных каналов SpaceWire, реализующих интерфейс дуплексных каналов связи (линков), которые могут функционировать со скоростью от 2 до 400 Мбит/с в каждую сторону. Независимая настройка скоростей передачи по линкам различных каналов. Скорости приема по линкам не зависят от скоростей передачи.
- МСК-01 осуществляет распределение меток времени, в соответствии со стандартом ECSS-E-50-12, а также кодов распределенных прерываний (в соответствии с проектом второй части международного стандарта SpaceWire.Part 2).
- МСК-01 имеет встроенный конфигурационный порт на базе процессора для обеспечения следующих функциональных возможностей: инициализации и конфигурирования коммутатора, выбора режима работы и управления функционированием, проведения мониторинга и диагностики состояния отдельного узла и сети SpaceWire в целом.
- Конфигурационный порт содержит блок внутренней системной памяти типа SRAM размером 16Кбайт (память программ), блок внутренней памяти типа SRAM размером 8 Кбайт (память пакетов) и блок внутренней памяти типа SRAM размером 1 Кбайт (таблица маршрутизации). Через параллельный 32-разрядный интерфейс имеется возможность подключения дополнительной системной памяти МСК-01. Имеется также возможность подключения внешнего процессора.
- Память программ конфигурационного порта предназначена для размещения встроенного ПО (firmware) маршрутизирующего коммутатора SpWitch-16 и не доступна для пользователей. Функции конфигурационного порта коммутатора реализуются программно встроенным процессором.
- Память пакетов предназначена для временного хранения пакетов, принимаемых из сети SpaceWire для конфигурационного порта и для пакетов, которые должны быть отправлены конфигурационным портом в сеть.

1.3 Структурная схема

Структурная схема МСК-01 приведена на Рисунок 1.1.

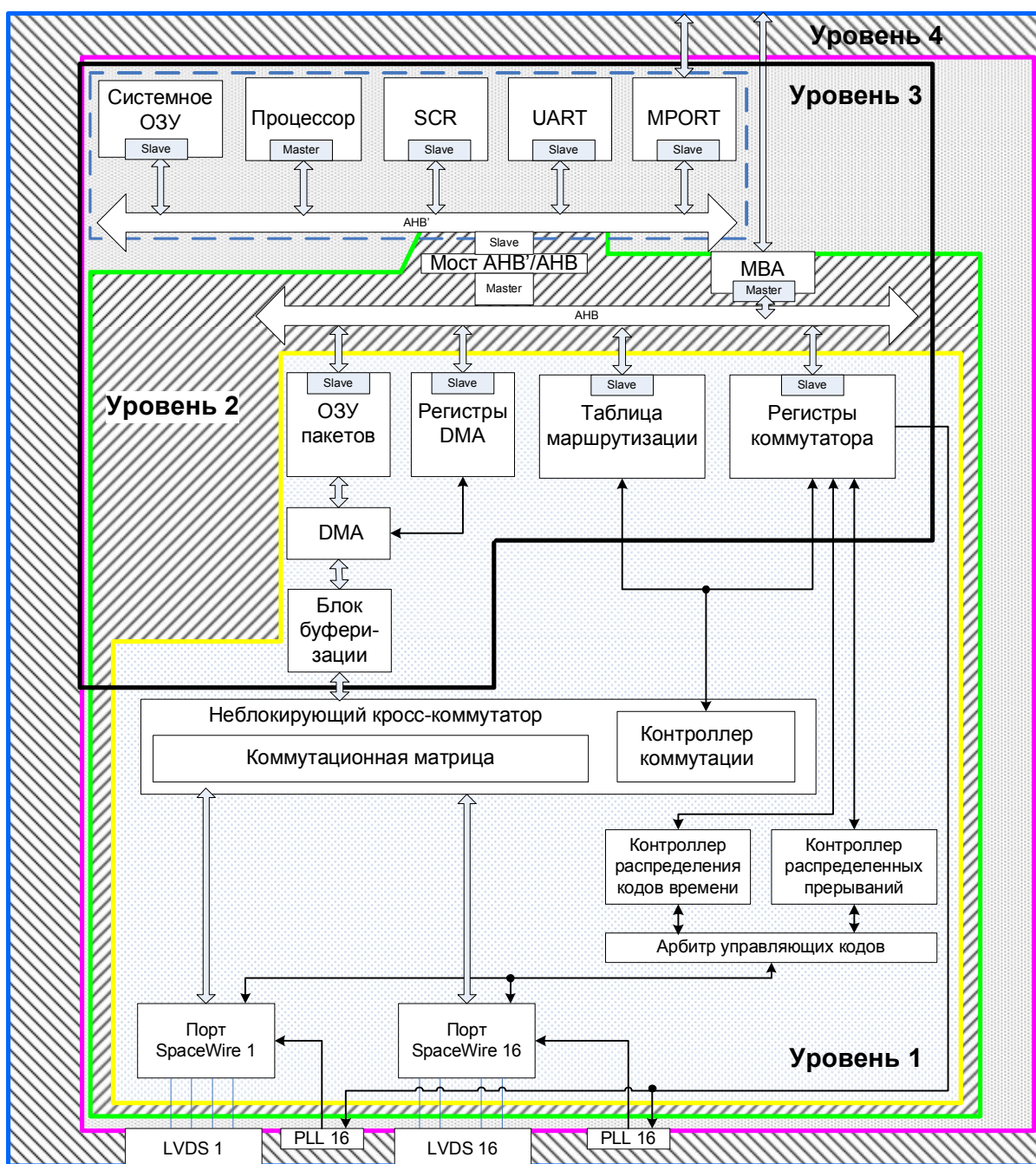


Рисунок 1.1 Структурная схема МСК-01

В состав МСК-01 входят следующие функциональные блоки:

- 16 портов SpaceWire (SWPORT 1, ..., SWPORT 16), реализующих интерфейс с линиями SpaceWire;
- неблокирующий кросс-коммутатор; данный компонент включает в себя: коммутационную матрицу для соединения приемных интерфейсов каналов SpaceWire с передающими интерфейсами и контроллер коммутации, управляющий ее функционированием (обеспечивает определение наиболее приоритетного среди поступивших пакетов и управление коммутацией при передаче

- пакетов между каналами SpaceWire с учетом возможностей групповой адаптивной маршрутизации);
- внутренний конфигурационный порт (порт 0) на базе встроенного процессора, доступный через кросс-коммутатор из каналов SpaceWire;
- таблица маршрутизации, доступная для записи через конфигурационный порт, которая обеспечивает отображение логического адреса на номер выходного порта SpaceWire;
- контроллер распределения управляющих кодов времени, необходимых для обеспечения синхронизации системного времени в процессорных модулях, являющихся терминальными модулями сети SpaceWire.
- контроллер распределенных прерываний, необходимых для обеспечения системных механизмов прерываний при организации распределенных вычислений;
- арбитр управляющих кодов, определяющий приоритет при выдаче управляющих кодов в каналы SpaceWire;
- блок регистров, доступных по записи и чтению через конфигурационный порт и содержащих управляющую информацию, необходимую для работы коммутатора в различных режимах, а также формирующих внешние сигналы состояния/ошибки для индикации рабочего и/или неисправного состояния каналов МСК-01; регистры используются встроенным ПО (firmware) МСК-01 и для пользователей недоступны.
- блок регистров DMA конфигурационного порта, которые доступны по записи и чтению через конфигурационный порт и которые содержат управляющую информацию, необходимую для записи в память пакетов, предназначенных для конфигурационного порта и чтения из памяти пакетов, предназначенных для отправки конфигурационным портом;
- системное ОЗУ, используемое как память программ после загрузки извне кода программы для встроенного процессора конфигурационного порта;
- ОЗУ пакетов, обеспечивающее буферизацию пакетов при их приеме и передаче из конфигурационного порта в сеть SpaceWire;
- внешний 32-разрядный параллельный порт (MPORT), доступный встроенному процессору для обращения к внешней системной памяти;
- внешний 32-разрядный параллельный порт (MBA), предназначенный для подключения к МСК-01 внешнего процессора;
- регистры управления CSR встроенного процессора;
- внешний порт JTAG, доступный встроенному процессору;
- UART, доступный встроенному процессору.

Структура МСК-01 представлена на четырех уровнях. Первый уровень включает в себя компоненты, осуществляющие собственно коммутацию.

Уровень 2 позволяет осуществлять подключение собственно коммутатора к внешнему процессору или через мост АНВ'/АНВ к встроенному процессору. Компонент уровня 2 является инвариантной частью с точки зрения реализации в ASIC и FPGA. Планируется, что в дальнейшем этот компонент будет инвариантной частью по отношению к различным будущим вариантам реализации коммутаторов SpaceWire, например, с конечным автоматом вместо встроенного процессора для упрощенных, компактных моделей коммутаторов.

В уровень 3 вынесены компоненты, которые необходимы для функционирования встроенного процессора или с которыми работать будет только этот процессор (UART).

На структурной схеме компоненты, входящие в конфигурационный порт, обведены жирной сплошной линией. Конфигурационный порт МСК-01 реализуется на базе

встроенного процессора и предназначен для обеспечения возможности инициализации и настройки конфигурации, управления режимами функционирования, проведения мониторинга и диагностики состояния отдельного узла и сети SpaceWire в целом. Также конфигурационный порт включает в себя параллельный 32-разрядный порт памяти (МВА), позволяющий подключение внешнего процессора.

Параметры конфигурации МСК-01 при проведении внешнего мониторинга доступны при обращении извне к конфигурационному порту (порту 0) через коммутационную матрицу. Регистры состояния МСК-01 и отдельных портов SpaceWire доступны только для чтения, регистры управления и таблица маршрутизации доступны для чтения и записи.

Программно управляемый конфигурационный порт позволяет обращаться к информации о конфигурации МСК-01 через любой из портов SpaceWire как в процессе инициализации системы, так и во время ее функционирования.

Конфигурационный порт, благодаря встроенному ПО (firmware), поддерживает реализацию различных протоколов конфигурации. Идентификатор протокола конфигурации используется процессором конфигурационного порта для определения и осуществления различных процедур управления МСК-01. Это обеспечивает возможность применения различных стратегий управления маршрутизирующими коммутаторами в сети SpaceWire – как централизованной, так и децентрализованной.

При централизованной стратегии конфигурирование каждого отдельного узла сети осуществляется сетевым администратором встроенного ПО (firmware) сети на базе МСК-01 из терминального узла, при этом обеспечивается реализация нескольких протоколов управления:

- настройка МСК-01 и статическая конфигурация таблицы маршрутизации;
- мониторинг и диагностика узлов сети SpaceWire;
- управления узлами сети SpaceWire.

Децентрализованный подход предполагает реализацию встроенным ПО (firmware) сети на базе МСК-01 дополнительных сетевых функций:

- динамическая настройка таблицы маршрутизации в МСК-01, что обеспечивает возможность «горячего» включения терминальных модулей;
- децентрализованная настройка максимально возможной скорости для каждого отдельного линка в МСК-01;
- автоматическая рассылка диагностических пакетов в случае выявления ошибок в линках SpaceWire и ошибок маршрутизации;
- автоматическое управление режимом экономии потребляемой мощности;
- управление ресурсами сети в соответствии с расширенным стандартом SpaceWire.

2. ПРОГРАММНАЯ МОДЕЛЬ

2.1 Общие положения

Управление МСК-01 осуществляется встроенным ПО (firmware) сети на базе МСК-01 через набор программно-доступных регистров. Чтение и запись регистров может быть осуществлена процессором МУЛЬТИКОР конфигурационного порта через интерфейс коммутатора шины AMBA АНВ.

2.2 Распределение адресного пространства

Распределение адресного пространства МСК-01 со стороны встроенного процессора интерфейса шины AMBA АНВ показано в Таблица 2.1.

Таблица 2.1 Распределение адресного пространства на шине AMBA АНВ

Начальный адрес	Конечный адрес	Реально используемый конечный адрес	Наименование блока
182F 5000	182F 53FC	182F 53FC	Таблица маршрутизации
182F 5400	182F 57FC	182F 5580	Регистры портов SpaceWire, управления коммутацией, контроллера распределения меток времени, контроллера распределенных прерываний
182F 5800	182F 5BFC	182F 5828	Регистры DMA
182F 8000	182F FFFC	182F FFFC	ОЗУ пакетов
1800 0000	1800 FFFC	1800 FFFC	Системное ОЗУ
182F 4000	182F 4FFC	182F 400C	CSR
182F 1000	182F 1FFC	182F 1018	MPORT
182F 3000	182F 3FFC	182F 3034	UART

Распределение адресного пространства МСК-01 со стороны интерфейса МВА показано в Таблица 2.2.

Таблица 2.2 Распределение адресного пространства со стороны интерфейса МВА

Начальный адрес	Конечный адрес	Реально используемый конечный адрес	Наименование блока
5000	53FC	53FC	Таблица маршрутизации
5400	57FC	5580	Регистры портов SpaceWire, управления коммутацией, контроллера распределения меток времени, контроллера распределенных прерываний
5800	5BFC	5828	Регистры DMA
8000	FFFC	FFFC	ОЗУ пакетов

2.3 Перечень регистров портов SpaceWire

Регистры доступны для встроенного процессора, а также через интерфейс МВА. Из неиспользуемых разрядов всех регистров считываются нули, при записи в них рекомендуется указывать нули.

Перечень программно-доступных для встроенного ПО регистров портов SpaceWire МСК-01 приведен в Таблица 2.3. В графе «адрес» указано смещение относительно 5400 – базового адреса начала адресного пространства регистров МСК-01.

Таблица 2.3 Перечень регистров состояния каналов SpaceWire

Условное обозначение	Описание	Тип доступа	Исходное состояние	Адрес
Status 1 [31...0]	Регистр статуса канала SpaceWire 1	WR/RD	0000	40
Status 2 [31...0]	Регистр статуса порта SpaceWire 2	WR/RD	0000	
Status 3 [31...0]	Регистр статуса порта SpaceWire 3	WR/RD	0000	
Status 4 [31...0]	Регистр статуса порта SpaceWire 4	WR/RD	0000	
Status 5 [31...0]	Регистр статуса порта SpaceWire 5	WR/RD	0000	
Status 6 [31...0]	Регистр статуса порта SpaceWire 6	WR/RD	0000	
Status 7 [31...0]	Регистр статуса порта SpaceWire 7	WR/RD	0000	
Status 8 [31...0]	Регистр статуса порта SpaceWire 8	WR/RD	0000	
Status 9 [31...0]	Регистр статуса порта SpaceWire 9	WR/RD	0000	
Status 10 [31...0]	Регистр статуса порта SpaceWire 10	WR/RD	0000	
Status 11 [31...0]	Регистр статуса порта SpaceWire 11	WR/RD	0000	
Status 12 [31...0]	Регистр статуса порта SpaceWire 12	WR/RD	0000	
Status 13 [31...0]	Регистр статуса порта SpaceWire 13	WR/RD	0000	
Status 14 [31...0]	Регистр статуса порта SpaceWire 14	WR/RD	0000	
Status 15 [31...0]	Регистр статуса порта SpaceWire 15	WR/RD	0000	7C
Status 16 [31...0]	Регистр статуса порта SpaceWire 16	WR/RD	0000	7C
MODE_CR1 [31...0]	Регистр режима работы порта SpaceWire 1	WR/RD	0009	80
MODE_CR2 [31...0]	Регистр режима работы порта SpaceWire 2	WR/RD	0009	
MODE_CR3 [31...0]	Регистр режима работы порта SpaceWire 3	WR/RD	0009	
MODE_CR4 [31...0]	Регистр режима работы порта SpaceWire 4	WR/RD	0009	
MODE_CR5 [31...0]	Регистр режима работы порта SpaceWire 5	WR/RD	0009	
MODE_CR6 [31...0]	Регистр режима работы порта SpaceWire 6	WR/RD	0009	
MODE_CR7 [31...0]	Регистр режима работы порта SpaceWire 7	WR/RD	0009	
MODE_CR8 [31...0]	Регистр режима работы порта SpaceWire 8	WR/RD	0009	
MODE_CR9 [31...0]	Регистр режима работы порта SpaceWire 9	WR/RD	0009	
MODE_CR10 [31...0]	Регистр режима работы порта SpaceWire 10	WR/RD	0009	
MODE_CR11 [31...0]	Регистр режима работы порта SpaceWire 11	WR/RD	0009	
MODE_CR12 [31...0]	Регистр режима работы порта SpaceWire 12	WR/RD	0009	
MODE_CR13 [31...0]	Регистр режима работы порта SpaceWire 13	WR/RD	0009	
MODE_CR14 [31...0]	Регистр режима работы порта SpaceWire 14	WR/RD	0009	
MODE_CR15 [31...0]	Регистр режима работы порта SpaceWire 15	WR/RD	0009	
MODE_CR16 [31...0]	Регистр режима работы порта SpaceWire 16	WR/RD	0009	BC
TX_SPEED1 [31...0]	Регистр коэффициента скорости передач порта SpaceWire 1	WR/RD	0000	C0
TX_SPEED2 [31...0]	Регистр коэффициента скорости передач порта SpaceWire 2	WR/RD	0000	
TX_SPEED3 [31...0]	Регистр коэффициента скорости передач порта SpaceWire 3	WR/RD	0000	
TX_SPEED4 [31...0]	Регистр коэффициента скорости передач порта SpaceWire 4	WR/RD	0000	
TX_SPEED5 [31...0]	Регистр коэффициента скорости передач порта SpaceWire 5	WR/RD R	0000	
TX_SPEED6 [31...0]	Регистр коэффициента скорости передач порта SpaceWire 6	WR/RD	0000	
TX_SPEED7 [31...0]	Регистр коэффициента скорости передач порта SpaceWire 7	WR/RD	0000	

Условное обозначение	Описание	Тип доступа	Исходное состояние	Адрес
TX_SPEED8 [31...0]	Регистр коэффициента скорости передач порта SpaceWire 8	WR/RD	0000	
TX_SPEED9 [31...0]	Регистр коэффициента скорости передач порта SpaceWire 9	WR/RD	0000	
TX_SPEED10 [31...0]	Регистр коэффициента скорости передач порта SpaceWire 10	WR/RD	0000	
TX_SPEED11 [31...0]	Регистр коэффициента скорости передач порта SpaceWire 11	WR/RD	0000	
TX_SPEED12 [31...0]	Регистр коэффициента скорости передач порта SpaceWire 12	WR/RD	0000	
TX_SPEED13 [31...0]	Регистр коэффициента скорости передач порта SpaceWire 13	WR/RD	0000	
TX_SPEED14 [31...0]	Регистр коэффициента скорости передач порта SpaceWire 14	WR/RD	0000	
TX_SPEED15 [31...0]	Регистр коэффициента скорости передач порта SpaceWire 15	WR/RD	0000	
TX_SPEED16 [31...0]	Регистр коэффициента скорости передач порта SpaceWire 16	WR/RD	0000	FC
RX_SPEED1 [7...0]	Регистр коэффициента скорости приема порта SpaceWire 1	RD	00	100
RX_SPEED2 [7...0]	Регистр коэффициента скорости приема порта SpaceWire 2	RD	00	
RX_SPEED3 [7...0]	Регистр коэффициента скорости приема порта SpaceWire 3	RD	00	
RX_SPEED4 [7...0]	Регистр коэффициента скорости приема порта SpaceWire 4	RD	00	
RX_SPEED5 [7...0]	Регистр коэффициента скорости приема порта SpaceWire 5	RD	00	
RX_SPEED6 [7...0]	Регистр коэффициента скорости приема порта SpaceWire 6	RD	00	
RX_SPEED7 [7...0]	Регистр коэффициента скорости приема порта SpaceWire 7	RD	00	
RX_SPEED8 [7...0]	Регистр коэффициента скорости приема порта SpaceWire 8	RD	00	
RX_SPEED9 [7...0]	Регистр коэффициента скорости приема порта SpaceWire 9	RD	00	
RX_SPEED10 [7...0]	Регистр коэффициента скорости приема порта SpaceWire 10	RD	00	
RX_SPEED11 [7...0]	Регистр коэффициента скорости приема порта SpaceWire 11	RD	00	
RX_SPEED12 [7...0]	Регистр коэффициента скорости приема порта SpaceWire 12	RD	00	
RX_SPEED13 [7...0]	Регистр коэффициента скорости приема порта SpaceWire 13	RD	00	
RX_SPEED14 [7...0]	Регистр коэффициента скорости приема порта SpaceWire 14	RD	00	
RX_SPEED15 [7...0]	Регистр коэффициента скорости приема порта SpaceWire 15	RD	00	
RX_SPEED16 [7...0]	Регистр коэффициента скорости приема порта SpaceWire 16	RD	00	13C

2.4 Описание регистров портов SpaceWire

2.4.1 Регистр статуса – Status

Адрес регистра определяется выражением: $(0x40) + (\text{номер_SpaceWire_канала} - 1) * 4$. Регистр статуса предназначен для оперативного контроля состояния фаз работы порта SpaceWire. Регистр доступен по чтению и записи. Запись в каждый отдельный разряд регистра выполняется по сигналам от DS-макроячейки. Сброс ряда разрядов регистра может осуществляться встроенным или внешним процессором путем записи в них '1'.

Назначение разрядов регистра STATUS показано в Таблица 2.4.

Таблица 2.4 Формат регистра STATUS

Номер разряда	Условное обозначение	Описание
0	DC_ERR	Признак ошибки разъединения (DisconnectError): '1' – ошибка произошла, '0' – нет ошибки (после сигнала сброса). Запись '1' в этот разряд сбрасывает этот разряд в '0'. После выхода МСК-01 или DS-макроячейки из состояния сброса этот разряд установлен в '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'. После выхода МСК-01 или DS-макроячейки из состояния сброса этот разряд установлен в '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'. После выхода МСК-01 или DS-макроячейки из состояния сброса этот разряд установлен в '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'. После выхода МСК-01 или DS-макроячейки из состояния сброса этот разряд установлен в '0'
4		Не используется
5...7	DS_STATE	Номер состояния, в котором в данный момент находится машина состояний DS-макроячейки: '000' – ErroReset – начальное состояние (состояние сброса), '001' – ErrorWait – ожидание возникновения ошибки, '010' – Ready – состояние готовности, '011' – Started – начало передачи, '100' – Connecting – ожидание кредитования, '101' – Run – передача данных. После выхода МСК-01 или DS-макроячейки из состояния сброса эти разряды установлены в '0'
8	BUFF_FULL	Устанавливается в '1', если буфер порта SpaceWire полон. После выхода МСК-01 или из состояния сброса этот разряд установлен в '0'
9		Не используется
10		Не используется
11	BUFF_EMPTY	Устанавливается в '1', если буфер порта SpaceWire пуст После выхода МСК-01 из состояния сброса этот разряд установлен в '0'
12	CONNECTED	Устанавливается в '1' при принятии первого бита при установке соединения. После выхода МСК-01 или DS-макроячейки из состояния сброса этот разряд ус-

Номер разряда	Условное обозначение	Описание
		тановлен в '0'
13...31	-	Не используется. Оставлено для будущих применений

2.4.2 Регистр режима работы – *MODE_CR*

Регистр режима работы доступен только по чтению. Формат регистра приведен в Таблица 2.5.

Таблица 2.5 Формат регистра *MODE_CR*

Номер разряда	Условное обозначение	Назначение
0	LinkDisabled	Установка LinkDisabled для блока DS-кодирования. При записи в этот разряд '1' управляющий сигнал LinkDisabled устанавливается в '1', при записи '0' – сбрасывается. После выхода МСК-01 из состояния сброса этот разряд установлен в '1'
1	AutoStart	Установка Autostart для блока DS-кодирования, при записи в этот разряд '1' управляющий сигнал Autostart устанавливается в '1', при записи '0' – сбрасывается. После выхода МСК-01 из состояния сброса этот разряд установлен в '0'
2	LinkStart	Установка LinkStart для блока DS-кодирования, при записи в этот разряд '1' управляющий сигнал LinkStart устанавливается в '1', при записи '0' – сбрасывается. После выхода МСК-01 из состояния сброса этот разряд установлен в '0'
3...4		Не используется
5	DS_RESET	Если этот разряд установлен в '0', то DS-макроячейка находится в состоянии сброса. После выхода МСК-01 из состояния сброса этот разряд установлен в '0'
6		Не используется
8		Не используется
9...10	-	Не используется
11	LVDS_LOOPBACK	При установке в '1' включается режим LVDS LoopBack. После выхода МСК-01 из состояния сброса этот разряд установлен в '0'
12	CODEC_LOOPBACK	При установке в '1' включается режим Codec LoopBack. После выхода МСК-01 из состояния сброса этот разряд установлен в '0'
13	BUF_MODE	Тип буферизации порта SpaceWire ('0' – запрос канала на передачу, если в буфере есть хотя бы один символ. '1' – запрос канала на передачу, если в буфере есть хотя бы один пакет или буфер полон). После выхода МСК-01 из состояния сброса этот разряд установлен в '0'

В начале работы и по сигналу сброса бит LinkDisabled устанавливается в '1', бит AutoStart='0' и LinkStart='0', DS_RESET='0'.

Для того чтобы DS-макроячейка корректно начала функционирование, необходимо сначала настроить соответствующую ей PLL, определяющую частоту передачи в канале на частоту 10 МГц. После этого можно однократной записью в регистр *MODE_CR* определить режим работы DS-макроячейки (LinkDisabled, AutoStart, LinkStart) и снять сигнал сброса, т. е. установить DS_RESET в '1', что обеспечит возможность установки соединения.

Соединение прекращается, если процессор осуществляет запись единицы в бит LinkDisabled либо DS_RESET.

2.4.3 Регистр коэффициента скорости передачи – *TX_SPEED*

Регистр коэффициента скорости передачи доступен по записи. Формат регистра показан в Таблица 2.6.

Таблица 2.6 Формат регистра *TX_SPEED*

Номер разряда	Условное обозначение	Назначение
0...7	TX_SPEED	Определяет скорость передачи данных
8...31	-	Резерв. Оставлено для будущих применений

2.4.4 Регистр коэффициента скорости приема – RX_SPEED

Восьмиразрядный регистр коэффициента скорости приема доступен по чтению. Значение регистра обновляется каждые 200 тактов HCLK (100 МГц) в соответствии с оценкой текущей скорости приема.

2.5 Перечень регистров управления

Перечень программно доступных регистров контроллера управления коммутацией и управляющих регистров МСК-01 приведен в Таблица 2.7.

Таблица 2.7 Перечень регистров управления коммутацией

Условное обозначение	Описание	Тип доступа	Исходное состояние	Адрес
ADG_ROUT_1 [15...0]	Регистр адаптивной групповой маршрутизации порта SpaceWire 1	WR/RD	0000	140
ADG_ROUT_2 [15...0]	Регистр адаптивной групповой маршрутизации порта SpaceWire 2	WR/RD	0000	
ADG_ROUT_3 [15...0]	Регистр адаптивной групповой маршрутизации порта SpaceWire 3	WR/RD	0000	
ADG_ROUT_4 [15...0]	Регистр адаптивной групповой маршрутизации порта SpaceWire 4	WR/RD	0000	
ADG_ROUT_5 [15...0]	Регистр адаптивной групповой маршрутизации порта SpaceWire 5	WR/RD	0000	
ADG_ROUT_6 [15...0]	Регистр адаптивной групповой маршрутизации порта SpaceWire 6	WR/RD	0000	
ADG_ROUT_7 [15...0]	Регистр адаптивной групповой маршрутизации порта SpaceWire 7	WR/RD	0000	
ADG_ROUT_8 [15...0]	Регистр адаптивной групповой маршрутизации порта SpaceWire 8	WR/RD	0000	
ADG_ROUT_9 [15...0]	Регистр адаптивной групповой маршрутизации порта SpaceWire 9	WR/RD	0000	
ADG_ROUT_10 [15...0]	Регистр адаптивной групповой маршрутизации порта SpaceWire 10	WR/RD	0000	
ADG_ROUT_11 [15...0]	Регистр адаптивной групповой маршрутизации порта SpaceWire 11	WR/RD	0000	
ADG_ROUT_12 [15...0]	Регистр адаптивной групповой маршрутизации порта SpaceWire 12	WR/RD	0000	
ADG_ROUT_13 [15...0]	Регистр адаптивной групповой маршрутизации порта SpaceWire 13	WR/RD	0000	
ADG_ROUT_14 [15...0]	Регистр адаптивной групповой маршрутизации порта SpaceWire 14	WR/RD	0000	
ADG_ROUT_15 [15...0]	Регистр адаптивной групповой маршрутизации порта SpaceWire 15	WR/RD	0000	
ADG_ROUT_16 [15...0]	Регистр адаптивной групповой маршрутизации порта SpaceWire 16	WR/RD	0000	17C
ID_VER[31...0]	Регистр версии МСК-01	RD	0002	0
ID_SWITCH [31...0]	Регистр идентификатора МСК-01	WR/RD	0000	4
SWITCH_CONTR	Регистр режима МСК-01	WR/RD	0000	8
ID_PROT[31...0]	Регистр идентификатора протокола	WR/RD	0000	C
ID_NET[15...0]	Регистр идентификации сетевых линков	WR/RD	0000	10

Условное обозначение	Описание	Тип доступа	Исходное состояние	Адрес
CONTROL_OUT [15...0]	Регистр выходного управляющего кода	WR/RD	0000	14
CUR_TIME[7...0]	Регистр текущего системного времени	WR/RD	0000	18
ISR_H[31...0]	Старшая половина регистра ISR	WR/RD	0000	1С
ISR_L[31...0]	Младшая половина регистра ISR	WR/RD	0000	20
Int_H_mack[31...0]	Старшая половина регистра маски распределенных прерываний	WR/RD	0000	24
Int_L_mack[31...0]	Младшая половина регистра маски распределенных прерываний	WR/RD	0000	28
Poll_H_mack[31...0]	Старшая половина регистра маски poll кодов	WR/RD	0000	2С
Poll_L_mack[31...0]	Младшая половина регистра маски poll кодов	WR/RD	0000	30
CUR_CONNECTED [15...0]	Регистр флагов установки соединения	WR/RD	0000	34
CUR_ERRORED [15...0]	Регистр флагов ошибок	WR/RD	0000	38
SWITCH_STATE	Регистр состояния MCK-01	WR/RD	0000	3С

2.6 Описание регистров управления

2.6.1 Регистр адаптивной групповой маршрутизации – ADG_ROUT

Регистр адаптивной групповой маршрутизации доступен процессору по чтению и записи. Регистр предназначен для хранения дополнительной информации об альтернативных линках для соответствующего порта SpaceWire. MCK-01 осуществляет групповую адаптивную маршрутизацию, управляемую от таблицы маршрутизации при использовании этой дополнительной информации (см. стандарт SpaceWire пп. 10.3.6). Формат регистра ADG_ROUT показан в Таблица 2.8.

Таблица 2.8 Назначение разрядов регистра ADG_ROUT

Номер разряда	Условное обозначение	Описание
0	ADG_ROUT1	Признак включения канала SpaceWire 1 в данную группу адаптивной маршрутизации: '1' – канал SpaceWire входит в состав группы; '0' – канал SpaceWire не входит в состав группы
1	ADG_ROUT 2	Признак включения канала SpaceWire 2 в данную группу адаптивной маршрутизации: '1' – канал SpaceWire входит в состав группы; '0' – канал SpaceWire не входит в состав группы
2	ADG_ROUT3	Признак включения канала SpaceWire 3 в данную группу адаптивной маршрутизации: '1' – канал SpaceWire входит в состав группы; '0' – канал SpaceWire не входит в состав группы
3	ADG_ROUT4	Признак включения канала SpaceWire 4 в данную группу адаптивной маршрутизации: '1' – канал SpaceWire входит в состав группы; '0' – канал SpaceWire не входит в состав группы
4	ADG_ROUT5	Признак включения канала SpaceWire 5 в данную группу адаптивной маршрутизации: '1' – канал SpaceWire входит в состав группы; '0' – канал SpaceWire не входит в состав группы
5	ADG_ROUT6	Признак включения канала SpaceWire 6 в данную группу адаптивной маршрутизации: '1' – канал SpaceWire входит в состав группы; '0' – канал SpaceWire не входит в состав группы

Номер разряда	Условное обозначение	Описание
6	ADG_ROUT7	Признак включения канала SpaceWire 7 в данную группу адаптивной маршрутизации: '1' – канал SpaceWire входит в состав группы; '0' – канал SpaceWire не входит в состав группы
7	ADG_ROUT8	Признак включения канала SpaceWire 8 в данную группу адаптивной маршрутизации: '1' – канал SpaceWire входит в состав группы; '0' – канал SpaceWire не входит в состав группы
8	ADG_ROUT9	Признак включения канала SpaceWire 9 в данную группу адаптивной маршрутизации: '1' – канал SpaceWire входит в состав группы; '0' – канал SpaceWire не входит в состав группы
9	ADG_ROUT10	Признак включения канала SpaceWire 10 в данную группу адаптивной маршрутизации: '1' – канал SpaceWire входит в состав группы; '0' – канал SpaceWire не входит в состав группы
10	ADG_ROUT11	Признак включения канала SpaceWire 11 в данную группу адаптивной маршрутизации: '1' – канал SpaceWire входит в состав группы; '0' – канал SpaceWire не входит в состав группы
11	ADG_ROUT12	Признак включения канала SpaceWire 12 в данную группу адаптивной маршрутизации: '1' – канал SpaceWire входит в состав группы; '0' – канал SpaceWire не входит в состав группы
12	ADG_ROUT13	Признак включения канала SpaceWire 13 в данную группу адаптивной маршрутизации: '1' – канал SpaceWire входит в состав группы; '0' – канал SpaceWire не входит в состав группы
13	ADG_ROUT14	Признак включения канала SpaceWire 14 в данную группу адаптивной маршрутизации: '1' – канал SpaceWire входит в состав группы; '0' – канал SpaceWire не входит в состав группы
14	ADG_ROUT15	Признак включения канала SpaceWire 15 в данную группу адаптивной маршрутизации: '1' – канал SpaceWire входит в состав группы; '0' – канал SpaceWire не входит в состав группы
15	ADG_ROUT16	Признак включения канала SpaceWire 16 в данную группу адаптивной маршрутизации: '1' – канал SpaceWire входит в состав группы; '0' – канал SpaceWire не входит в состав группы
16...31	-	Резерв. Оставлено для будущих применений. Содержит '0'

Регистр содержит суперпозицию унитарных кодов номеров портов SpaceWire, альтернативных данному порту, указанному в таблице маршрутизации. Групповая адаптивная маршрутизация позволяет направлять пакет по одному из ряда альтернативных каналов, соединяющих смежные коммутаторы и/или терминальные узлы. Групповая адаптивная маршрутизация помогает обеспечивать поддержку для совместного использования пропускной способности каналов и/или отказоустойчивости в сети SpaceWire. Начальное значение всех разрядов регистра адаптивной групповой маршрутизации после выхода МСК-01 из состояния сброса – '0'.

2.6.2 Регистр идентификатора МСК-01–ID_SWITCH

32-разрядный регистр идентификатора МСК-01 реализован с доступом по чтению и записи. Регистр может быть запрограммирован через конфигурационный порт на зна-

чение идентификации данного коммутатора или другую информацию, чтобы поддерживать алгоритмы исследования сети.

2.6.3 Регистр режима работы МСК-01 – SWITCH_CONTR

Регистр режима работы МСК-01 реализован с доступом по чтению и записи. Назначение разрядов регистра приведено в Таблица 2.9.

Таблица 2.9 Назначение разрядов регистра SWITCH_CONTR

Номер разряда	Условное обозначение	Описание
0...5	BaseTime	Базовое значение длительности интервала между последовательными сменами приоритетов каналов. После выхода МСК-01 из состояния сброса значение этих разрядов '000000'. (В этом случае смена приоритетов будет осуществляться 1 разв 16 тактов)
6	TcodeMack	Маска timescode – если этот разряд установлен в '1', то при приходе корректного маркера времени прерывание IRQ2 не устанавливается. После выхода МСК-01 из состояния сброса значение этого разряда – '0'
7	RSTIRQ2	При записи '1' в этот разряд осуществляется сброс прерывания IRQ2 После выхода МСК-01 из состояния сброса значение этого разряда– '0'
8...11	IRQMack	Маска для формирования прерывания IRQ для внешнего процессора. Если разряд 8 установлен в '1', то в формировании IRQ не участвует IRQ0; если разряд 9 установлен в '1', то в формировании IRQ не участвует IRQ1; если разряд 10 установлен в '1', то в формировании IRQ не участвует IRQ2; если разряд 11 установлен в '1', то в формировании IRQ не участвует IRQ3. После выхода МСК-01 из состояния сброса значение этих разрядов '0'
27...12	ERRORMack	Маска для установки сигнала ERROR, если j-ый бит маски установлен в '1', то возникновение ошибки в j-ой DS-макроячейке не служит причиной для установки сигнала ERROR. После выхода МСК-01 из состояния сброса значение этих разрядов '0'
31...28	DisTime	Смещение для базового значения интервала между последовательными сменами приоритетов каналов. После выхода МСК-01 из состояния сброса значение этих разрядов '0'

2.6.4 Регистр идентификатора протокола – ID_PROT

32-разрядный регистр идентификатора МСК-01 реализован с доступом по чтению и записи. Регистр может быть запрограммирован через конфигурационный порт на значение идентификатора номера протокола, который поддерживается конфигурационным портом МСК-01. В зависимости от типа протокола, могут изменяться алгоритмы интерпретации в контроллере управления коммутацией заголовка пакета, формируемого в конфигурационном порту при мониторинге состояния узлов сети или при изменении их состояния.

После выхода МСК-01 из состояния сброса значение этого регистра –'0'.

2.6.5 Регистр идентификации сетевых линков – ID_NET

16-разрядный регистр идентификации сетевых линков реализован с доступом по чтению и записи. Если к i-му порту SpaceWire подключен терминальный узел, то разряд i этого регистра рекомендуется устанавливать в '0', если к этому порту подключен порт другого коммутатора, то разряд i рекомендуется устанавливать в '1'.Если в i разряде

этого регистра установлен '0', то для порта SpaceWire *i* разрешено ширококешание. Если в разряде *i* этого регистра установлен '1', то для *i*-го порта SpaceWire запрещено ширококешание, т. е. пакеты, адресованные более чем одному каналу (группе каналов) в данный порт передаваться не будут.

После выхода МСК-01 из состояния сброса значение всех разрядов этого регистра – '0'.

2.6.6 Регистр выходного управляющего кода – CONTROL_OUT

Восьмиразрядный регистр выходного управляющего кода реализован с доступом по чтению и записи. Данный регистр может быть использован встроенным процессором (внешним процессором, подключенным через интерфейс MBA) для отправки в сеть маркера времени, кода распределенного прерывания или poll кода. Как только встроенный (внешний) процессор осуществляет запись в этот регистр, записанный управляющий код поступает в контроллер обработки управляющих кодов времени или контроллер обработки распределенных прерываний.

После выхода МСК-01 из состояния сброса значение разрядов этого регистра – '0'.

2.6.7 Регистр текущего системного времени – CUR_TIME

Шестиразрядный регистр текущего системного времени реализован с доступом по чтению. Данный регистр содержит значение текущего системного времени.

После выхода МСК-01 из состояния сброса значение разрядов этого регистра – '0'.

2.6.8 Регистр ISR_H, L

Регистры ISR_H[31...0], ISR_L[31...0] реализованы с доступом по чтению. Данные регистры содержат значения флагов распространения распределенных прерываний из регистра ISR[63...0]. Если в *i* разряде регистра ISR '1' – флаг установлен, что означает фиксацию факта прохождения через данный МСК-01 кода распределенного прерывания со значением, равным двоичному коду номера *i*; если '0' – флаг сброшен при приходе управляющего кода poll со значением, равным двоичному коду номера *i*.

После выхода МСК-01 из состояния сброса значение всех разрядов этого регистра – '0'.

2.6.9 Регистр маски распределенных прерываний – Int_H, L_mask

Регистры Int_H_mask[31...0], Int_L_mask[31...0] реализованы с доступом по чтению и записи. Данные регистры предназначены для определения маски распределенных прерываний (определяют, при получении коммутатором каких распределенных прерываний будет установлено прерывание IRQ2 для встроенного процессора). Если в *i* разряде '0' – прерывание при приходе кода распределенного прерывания с номером *i* разрешено, если '1' – запрещено.

После выхода МСК-01 из состояния сброса значение всех разрядов этого регистра – '0'.

2.6.10 Регистр маски poll кодов – Poll_H, L_mask

Регистры Poll_H_mask[31...0], Poll_L_mask[31...0] реализованы с доступом по чтению и записи. Данные регистры предназначены для определения маски распределенных прерываний (определяют, при получении коммутатором каких распределенных прерываний будет установлено прерывание IRQ2 для внутреннего процессора). Если в *i* разряде '0' – прерывание при приходе poll кода с номером *i* разрешено, если '1' – запрещено.

После выхода МСК-01 из состояния сброса значение всех разрядов этого регистра – '0'.

2.6.11 Регистр флагов установки соединения – CUR_CONNECTED

16-разрядный регистр флагов установки соединения реализован с доступом по чтению. Если бит *i* этого регистра установлен в '1', то по каналу SpaceWire МСК-01 с номером *i* в текущий момент времени установлено соединение.

После выхода МСК-01 из состояния сброса значение всех разрядов этого регистра – '0',.

2.6.12 Регистр флагов ошибок – *CUR_ERRORED*

16-разрядный регистр флагов ошибок реализован с доступом по чтению. Если бит *i* этого регистра установлен в '1', то по каналу SpaceWire МСК-01 с номером *i* соединение в текущий момент времени разорвано вследствие ошибки.

После выхода МСК-01 из состояния сброса значение всех разрядов этого регистра – '0'.

2.6.13 Регистр состояния МСК-01 – *SWITCH_STATE*

Регистр состояния МСК-01 реализован с доступом по чтению и по записи. Назначение битов этого регистра приведено в Таблица 2.10.

Таблица 2.10 Назначение разрядов регистра *SWITCH_STATE*

Номер разряда	Условное обозначение	Описание
3...0	IRQ3, IRQ2, IRQ1, RQ0	В соответствующие разряды отображается значение сигналов прерываний IRQ3, IRQ2, IRQ1, RQ0. После выхода МСК-01 из состояния сброса значение этих разрядов – '0'
4	STATUSbit	Бит статуса, его значение отображается на выход STATUS МСК-01. Назначение бита определяется программно, путем записи в соответствующий разряд. После выхода МСК-01 из состояния сброса значение этого разряда – '0'
31...5		Назначение разрядов определяется программно. После выхода МСК-01 из состояния сброса значение этих разрядов – '0'

2.7 Регистры DMA

В Таблица 2.11 приведены относительные адреса регистров.

Таблица 2.11 Описание регистров DMA

Номер регистра	Адрес регистра	Тип доступа	Описание регистра
REG_RX_DATA_ADDR	0	WR/RD	Регистр адреса области DATA на прием
REG_TX_DATA_ADDR	4	WR/RD	Регистр адреса области DATA на передачу
REG_RX_DESC_ADDR	8	WR/RD	Регистр адреса области DESC на прием
REG_TX_DESC_ADDR	C	WR/RD	Регистр адреса области DESC на передачу
REG_RX_DATA_LEN	10	WR	Регистр длины области DATA на прием
REG_TX_DATA_LEN	14	WR	Регистр длины области DATA на передачу
REG_RX_DESC_LEN	18	WR	Регистр длины области DESC на прием
REG_TX_DESC_LEN	1C	WR	Регистр длины области DESC на передачу
REG_DMA_CONTROL_AND_STATE	20	WR/RD	Регистр управления и состояния DMA
REG_DMA_MAX_TRAN	24	WR/RD	Регистр размера максимальной транзакции

В Таблица 2.12 приводятся номера и описания разрядов регистра управления и состояния REG_DMA_CONTROL_AND_STATE.

Таблица 2.12 Описание разрядов регистра управления и состояния DMA

Номер разряда	Номер по умолчанию	Тип доступа	Описание регистра
BIT_RX_DATA_CONTROL	0	WR/RD	'1' – работа канала приема данных разрешена

BIT_TX_DATA_CONTROL	1	WR/RD	'1' – работа канала передачи данных разрешена
BIT_RX_DESC_CONTROL	2	WR/RD	'1' – работа канала приема данных разрешена
BIT_TX_DESC_CONTROL	3	WR/RD	'1' – работа канала приема данных разрешена
BIT_RX_DATA_IRQ	4	RD	'1' – требуется настройка регистров канала приема данных
BIT_TX_DATA_IRQ	5	RD	'1' – требуется настройка регистров канала передачи данных
BIT_RX_DESC_IRQ	6	RD	'1' – требуется настройка регистров канала приема дескрипторов
BIT_TX_DESC_IRQ	7	RD	'1' – требуется настройка регистров канала передачи дескрипторов
BIT_RX_DATA_WN	8	WR/RD	'1' – режим группового обмена с памятью. Размер транзакции определяется в регистре REG_DMA_MAX_TRAN. '0' – режим однословного обмена
BIT_TX_DATA_WN	9	WR/RD	'1' – режим группового обмена с памятью. Размер транзакции определяется в регистре REG_DMA_MAX_TRAN. '0' – режим однословного обмена
BIT_RX_DESC_WN	10	WR/RD	'1' – режим группового обмена с памятью. Размер транзакции определяется в регистре REG_DMA_MAX_TRAN. '0' – режим однословного обмена
BIT_TX_DESC_WN	11	WR/RD	'1' – режим группового обмена с памятью. Размер транзакции определяется в регистре REG_DMA_MAX_TRAN. '0' – режим однословного обмена

2.8 Формат таблицы маршрутизации

Таблица маршрутизации содержит отображение логических адресов пакетов на физические адреса (номера) каналов SpaceWire в пределах МСК-01. Распределение адресов в таблице маршрутизации МСК-01 показано в Таблица 2.13.

Таблица 2.13 Распределение адресов в таблице маршрутизации МСК-01

Диапазон адресов	Функция
0	Внутренний конфигурационный порт
1...31 (01...1F hex)	Физические выходные порты SpaceWire
32...254 (20...FF hex)	Логические адреса, которые отображаются на физические выходные порты

Пример таблицы маршрутизации приведен в Таблица 2.14.

Таблица 2.14 Пример таблицы маршрутизации для 16-портового маршрутизатора

Функция	Адрес	Порты						Приоритет	Признак удаления заголовка
		0	1	2	3	...	16		
Конфигурация	0	1	0	0	0		0	0	1
Адресация пути	1	0	1	0	0		0	0	1
	2	0	0	1	0		0	0	1
	3	0	0	0	1		0	0	1
	...								1
	16	0	0	0	0		1	1	1
Логическая адресация	...								1
	32	0	0	1	0		0	1	0
	33	0	0	0	0		1	0	0
	34	0	1	0	0		0	1	0
Резерв	255	0	0	0	0		0	0	0

Внутренний конфигурационный порт используется для доступа к таблице маршрутизации и другой информации о конфигурации, проводимой в МСК-01.

2.9 Описание процесса обработки управляющих кодов времени в МСК-01

МСК-01 обеспечивает распространение по сети управляющих кодов времени в соответствии со стандартом SpaceWire. Вновь поступивший код времени считается корректным, если его значение на '1' больше значения предыдущего кода времени (либо значение предыдущего маркера времени '63', а значение нового – '0'). МСК-01 распространяет корректный код времени по сети. Если же поступает некорректный управляющий код времени, он фиксируется в МСК-01, но дальше по сети не распространяется.

Коды времени могут поступать в МСК-01 по всем каналам SpaceWire. Значение вновь поступившего кода времени сравнивается со значением регистра текущего системного времени CUR_TIME. Если код времени корректен, то он распространяется дальше по сети. Множество каналов SpaceWire, по которым в этом случае будет рассылаться код времени, определяется следующим образом. Код времени не отсылается в канал, по которому он поступил, а также в каналы, альтернативные порту, по которому он поступил. Множество этих каналов определяется в соответствии со значением регистра ADG_ROUT_i, где *i* – номер порта, по которому поступил код времени. Код времени рассылается в остальные каналы таким образом, чтобы в каждой группе альтернативных каналов код времени был отправлен только по одному из них, если в группе имеется хотя бы один работоспособный канал.

Если значение вновь поступившего кода времени не корректно, то он не рассылается по выходным портам коммутатора.

Значение кода времени в любом случае записывается в регистр CUR_TIME.

Если в МСК-01 в течение малого промежутка времени (нижняя граница этой задержки равна 35 нс и определяется временем передачи 14-разрядного кода времени по линку SpaceWire со скоростью 400 Мбит/с) поступило несколько кодов времени, то обработка этих кодов времени осуществляется в порядке номеров каналов, по которым они поступили – от меньшего к большему. В коммутатор может практически одновременно поступить несколько кодов времени, имеющих одно и то же значение. Это может произойти, если в системе существует несколько различных путей между источником кодов времени и МСК-01. В этом случае нет принципиальной разницы, в каком порядке будут обрабатываться данные коды времени. Если код времени будет послан в канал, по которому уже был принят код времени с таким же значением (но еще не обработан), то его дальнейшее распространение будет прекращено узлом, в который он поступит.

При корректном проектировании сети SpaceWire и системы в целом должна быть исключена ситуация, когда в коммутатор практически одновременно поступают коды времени *i* и *i-1* (код *i-1* мог распространяться в сети по более длинному пути). Это означает, что при проектировании механизма распределения системного времени интервалы между поступлениями кодов времени из модуля – источника в сеть SpaceWire должны выбираться таким образом, чтобы обеспечить распространение в сети SpaceWire в один момент времени только одного кода времени *i*. Это условие будет обеспечиваться, если период генерации кодов времени будет больше времени распространения кода по пути, равного диаметру сети (т. е. наибольшему из всех кратчайших маршрутов между всеми парами терминальных узлов).

Значение текущего системного времени хранится в регистре CUR_TIME, который доступен по чтению как встроенному процессору МСК-01, так и внешнему процессору, подключаемому к МСК-01. Если в регистре режима работы МСК-01 SWITCH_CONTR не замаскирована установка прерывания IRQ[2] при поступлении очередного корректного кода времени, это прерывание может быть использовано встроенным (внешним) процессором для отслеживания факта приема корректного кода времени.

Встроенный процессор МСК-01, а также внешний процессор могут выступать в качестве источников кодов времени. Для отправки кода времени в сеть его значение необходимо записать в регистр CONTROL_OUT.

2.10 Описание процесса обработки кодов распределенных прерываний и poll кодов в МСК-01

МСК-01 обеспечивает распространение по сети SpaceWire кодов распределенных прерываний и poll кодов в соответствии с проектом второй очереди стандарта SpaceWire. Факты поступления кодов распределенных прерываний и poll кодов регистрируются в регистре ISR коммутатора. На основе информации, хранящейся в этом регистре, определяется, будет ли вновь поступивший код распределенного прерывания или poll код отправлен далее по сети SpaceWire.

Если в МСК-01 поступает код распределенного прерывания со значением i и соответствующий разряд регистра $ISR[i]=0$, то данный код распределенного прерывания рассылается далее по сети. $ISR[i]$ в этом случае устанавливается в '1'. Если же $ISR[i]$ уже был установлен в '1', то поступивший код распределенного прерывания игнорируется. Этот механизм обеспечивает отсеивание копий одного и того же кода распределенного прерывания, поступивших в МСК-01 по разным маршрутам. (В корректно спроектированной сети должен быть только один источник распределенных прерываний каждого типа. Корректно функционирующий источник распределенных прерываний отправляет в сеть следующий код распределенного прерывания i только после того, как получит poll код i , либо после истечения времени ожидания poll кода i .)

Если в МСК-01 поступает poll код со значением i и $ISR[i]=1$, то данный poll код рассылается далее по сети. $ISR[i]$ в этом случае устанавливается в '0'. Если же $ISR[i]$ уже был установлен в '0', то поступивший poll код игнорируется. Этот механизм обеспечивает отсеивание копий poll кода, поступивших в МСК-01 по разным маршрутам.

Множество каналов SpaceWire, по которым будет рассылаться код распределенного прерывания или poll код, определяется следующим образом. Код распределенного прерывания (poll код) не отсылается в канал, по которому он поступил, а также в каналы, альтернативные порту, по которому он поступил. Множество этих каналов определяется в соответствии со значением регистра ADG_ROUT_i , где i – номер порта, по которому поступил управляющий код. Код распределенного прерывания (poll код) рассылается в остальные каналы таким образом, чтобы в каждой группе альтернативных каналов управляющий код был отправлен только по одному из них, если в группе имеется хотя бы один работоспособный канал.

Коды распределенных прерываний и poll коды могут поступать в МСК-01 по всем портам SpaceWire. Для каждого порта существует отдельный регистр, в котором фиксируется значение поступившего кода распределенного прерывания (poll кода). Обработка поступающих кодов распределенных прерываний (poll кодов) от портов SpaceWire организована в соответствии со схемой циклических приоритетов. Регистрация в регистре ISR поступления кода распределенного прерывания (poll кода) осуществляется за один такт локальной частоты работы МСК-01 (10 нс).

Для того чтобы гарантированно не произошла утрата кода распределенного прерывания (poll кода) в результате его перезаписи необходимо, чтобы по одному каналу SpaceWire коды распределенных прерываний (poll коды) поступали не чаще, чем 1 раз в 160 нс (в 16 тактов локальной частоты работы МСК-01).

Если в МСК-01 значение одного и того же кода распределенного прерывания поступит в течение небольшого интервала времени по нескольким каналам SpaceWire (в сети между источником распределенных прерываний и МСК-01 существует несколько путей почти одинаковой длины), то не исключена ситуация, когда код распределенного прерывания (poll код) будет отправлен по каналу, по которому уже был получен код с

таким же значением. Эта ситуация не является критичной для сети, поскольку такой код будет проигнорирован получившим его коммутатором или терминальным узлом. Встроенный процессор МСК-01, как и внешний процессор, может прочитать значение регистра ISR, а также может выступать в качестве источника распределенных прерываний. Для того чтобы отправить распределенное прерывание в сеть, необходимо записать его значение в регистр CONTROL_OUT.

Встроенный (внешний) процессор МСК-01 может выступать в качестве обработчика распределенных прерываний (источника poll кодов). Для того чтобы отправить poll код в сеть, необходимо записать его значение в регистр CONTROL_OUT.

Факт приема распределенного прерывания (poll кода) из сети может быть определен процессором по установке прерывания IRQ[2], если соответствующее распределенное прерывание (poll код) не замаскировано в регистре маски Int_H,L_mask (Poll_H,L_mask).

2.11 Описание процесса обработки пакетов данных в МСК-01

Пакеты данных могут поступать в МСК-01 по всем каналам SpaceWire. Первый байт пакета (байт, пришедший вслед за очередным концом пакета) рассматривается как заголовок, по которому определяется, в какие каналы SpaceWire этот пакет будет отправлен. Если вслед за очередным символом конца пакета вновь поступает символ конца пакета, то последний символ конца пакета отбрасывается.

В заголовке каждого пакета, поступающего в коммутатор, содержится двоичный код номера порта назначения либо логический адрес терминального узла назначения. Каналы МСК-01, по которым будет отправлен пакет, определяются на основе заголовка пакета, информации в таблице маршрутизации, регистра идентификации сетевых линков, регистров адаптивной групповой маршрутизации и состояния выходных портов SpaceWire.

Заголовок пакета используется в качестве адреса в таблице маршрутизации, по которому определяется базовый набор портов SpaceWire, в которые должен быть разослан пакет, приоритет пакета, а также, должен ли в коммутаторе быть удален заголовок.

Пусть, например, в коммутатор поступил пакет со значением заголовка 35. Этому заголовку соответствует строка 35 в таблице маршрутизации, которая содержит информацию, показанную на Рисунке 1.1.

Номер порта	31			18 17	15	11 8	7 5	3 1
Строка таблицы маршрутизации	0000	0000	0000	0000	0000	0000	0010	1010

Бит удаления заголовка

Бит приоритета

Рисунок 2.1 Пример строки таблицы маршрутизации

В разряде 17 стоит '0' – приоритет пакета равен '0'. В разряде 18 тоже '0' – заголовок пакета не должен удаляться.

В разрядах 1, 3, 5 стоят '1', соответственно базовый набор портов, в которые должен быть разослан данный пакет – 1, 3, 5. В первую очередь строка таблицы маршрутизации анализируется на количество '1' в разрядах 0...16 слова, чтобы определить широковещательная или единичная передача пакета имеет место. Если в строке более одной '1', что соответствует широковещательной передаче, то используются данные из регистра идентификации сетевых линков в качестве маски. Цель этого маскирования заключается в том, чтобы оставить только те порты SpaceWire, к которым подключены терминальные узлы. В соответствии со стандартом SpaceWire, маршрутизирующий коммутатор может использовать режим шировещания для передачи пакета только

этим узлам. Это позволяет исключить риск блокировки коммутаторов, использующих маршрутизацию типа «wormhole» при передаче пакета через сеть SpaceWire.

Если в базовом наборе ко всем выделенным портам (1, 3, и 5) подключены терминальные узлы, то полученный таким образом набор выходных портов SpaceWire может быть скорректирован с учетом регистров адаптивной групповой маршрутизации. В соответствии со значениями регистров ADG_ROUT1, ADG_ROUT3 и ADG_ROUT5 определяется фактический набор каналов, по которому будет разослан данный пакет.

Пусть, например,

ADG_ROUT1= 0000 0000 0000 0000 0000 0000 0000 0010

ADG_ROUT3= 0000 0000 0000 0000 0000 0000 0000 1100

ADG_ROUT5= 0000 0000 0000 0000 0000 0000 1111 0000

В соответствии с этим группа альтернативных каналов для порта 1 включает только этот канал. Группа альтернативных каналов для порта 3 включает канал 2 и канал 3.

Группа альтернативных каналов для порта 5 включает в себя каналы 4, 5, 6, 7.

При выборе в группе канала, по которому будет фактически отправлен пакет, сначала отбираются все исправные каналы, затем среди них все свободные. Среди них выбирается канал с наименьшим номером.

Пусть, например, в текущий момент времени в группе альтернативных каналов порта 3 все исправны и свободны. В этом случае среди них будет отобран канал 2. Пусть в группе для порта 5 канал 4 занят, канал 5 неисправен, а каналы 6 и 7 свободны. В этом случае среди них будет отобран канал 6.

Таким образом, рассматриваемый в примере пакет будет разослан в канал 1 (независимо от его состояния, поскольку для него альтернативные каналы не определены), канал 2 и канал 6.

Если номер порта, которому адресован пакет данных, равен '0', то данный пакет поступит в конфигурационный порт и будет записан в память пакетов в соответствии с настройками DMA. Из памяти пакетов в дальнейшем он может быть прочитан встроенным или внешним процессором.

Если пакет отсылается в сеть встроенным или внешним процессором, его заголовок является не адресом в строке таблицы маршрутизации, а строкой, имеющей такой же формат, как и строка таблицы маршрутизации (и имеет длину не один, а четыре байта соответственно). Поэтому при отправке пакета от конфигурационного порта в сеть чтение таблицы маршрутизации не выполняется, обработка заголовка пакета осуществляется аналогично обработке строки таблицы маршрутизации.

Если пакет адресован неисправному каналу или каналу, по которому в данный момент не установлено соединение, что зафиксировано в соответствующем разряде регистра CUR_CONNECTED, для которого не определены альтернативные каналы, или все его альтернативные каналы неисправны, то пакет изымается из сети.

Если пакет адресован группе каналов, среди которых есть неисправные (и для этих неисправных каналов нет исправных альтернативных каналов), данный пакет рассылается только тем каналам из группы, которые исправны.

Отправка пакета, адресованного группе каналов, осуществляется следующим образом. Когда все порты SpaceWire подтвердили готовность принять очередной байт, он передается всем каналам. Таким образом, передача пакета, адресованного группе каналов, осуществляется на скорости самого медленного канала из группы.

2.12 Описание логики работы прерываний в МСК-01

В МСК-01 формируется 4 прерывания для встроенного процессора и одно прерывание для внешнего процессора. Внутренние прерывания МСК-01: IRQ0 – прерывание устанавливается при установке соединения, IRQ1 – прерывание устанавливается при разрыве соединения, IRQ2 – прерывание устанавливается при получении управляющего кода из сети, IRQ3 – прерывание от DMA конфигурационного порта.

После снятия сигнала сброса все сигналы прерываний установлены в '0' (неактивное состояние). Как только по какому-либо из каналов SpaceWire происходит установка соединения (машина состояний DS-макроячейки порта SpaceWire переходит в состояние run), сигнал прерывания IRQ0 устанавливается в '1'. Для того чтобы произошел сброс сигнала прерывания IRQ0 необходимо произвести запись '1' в разряд 12 регистра состояния канала SpaceWire (Status i), по которому было установлено соединение. Если на момент записи в регистр состояния, соединение было установлено не только по данному каналу, но и по другим каналам, сброса сигнала прерывания IRQ0 не произойдет. Сигнал будет оставаться в активном состоянии до тех пор, пока не будет осуществлена запись в регистры состояния всех каналов SpaceWire, по которым было установлено соединение. Если в канале SpaceWire происходит разрыв и повторная установка соединения (и сигнал прерывания находился в неактивном состоянии), то прерывание IRQ0 будет установлено повторно.

Прерывание IRQ1 устанавливается в '1', если по одному (или нескольким) каналам происходит разрыв соединения вследствие внешних причин. Если разрыв соединения происходит вследствие программного сброса порта SpaceWire по инициативе встроенного или внешнего процессора, то данное прерывание не устанавливается. Данное прерывание может быть сброшено программно или аппаратно. Для программного сброса необходимо осуществить запись '1' в разряды 3...0 регистра состояния (Status i), можно осуществлять запись '1' только в те разряды, которые установлены в '1'.

Прерывание IRQ1 будет сброшено аппаратно, если по каналу произошла повторная установка соединения.

Если разрыв соединения произошел по нескольким каналам, прерывание IRQ1 будет сброшено только после того, как будет программно или аппаратно устранена причина установки прерывания по всем этим каналам.

Прерывание IRQ2 может быть установлено, если из сети принят очередной корректный маркер времени, код распределенного прерывания или roll код. Возможно маскирование каждой из причин данного прерывания. Для того чтобы прерывание не устанавливалось при приходе корректных маркеров времени необходимо в разряд 6 регистра режима работы коммутатора (SWITCH_CONTR) записать значение '1'. Для того чтобы прерывание не устанавливалось при получении конкретного кода распределенного прерывания или roll кода, необходимо соответствующий разряд маски установить в '1' (Int_H_mack, Int_L_mack, Poll_H_mack, Poll_L_mack).

После сброса коммутатора ни одна из причин возникновения IRQ2 не является замаскированной.

Для того чтобы сбросить IRQ2, необходимо в разряд 6 регистра управления коммутатора (SWITCH_CONTR) записать '1'.

Прерывание IRQ3 устанавливается DMA конфигурационного порта, если чтение из памяти разрешено и при этом закончилась область данных или область дескрипторов, выделенная для чтения, и/или если запись из памяти разрешена и при этом закончилась область данных или область дескрипторов, выделенная для записи. Сброс данного прерывания осуществляется после того, как DMA выделена новая область данных и/или дескрипторов.

Прерывание для внешнего процессора формируется комбинаторно (логика «ИЛИ») на базе значений сигналов прерывания для встроенного процессора. Для того чтобы IRQ0, IRQ1, IRQ2 и/или IRQ3 не участвовал в формировании прерывания для внешнего процессора в разряд регистра режима коммутатора (SWITCH_CONTR) 8, 9, 10, 11 соответственно необходимо записать '1'. После снятия сигнала сброса эти разряды установлены в '0'.

Для ускорения процесса обработки прерывания внешним процессором, а также для того, чтобы внешний процессор мог осуществлять работу в режиме мониторинга значе-

ния сигналов прерываний IRQ0, IRQ1, IRQ2, IRQ3 отображаются в регистре состояния коммутатора (SWITCH_STATE), разряды 0, 1, 2, 3 соответственно.

3. РЕКОМЕНДАЦИИ ПО ПРОГРАММИРОВАНИЮ

МСК-01 будет поставляться со встроенным ПО (firmware) конфигурационного порта.. Программирование МСК-01 пользователями СБИС не предполагается.

В начале работы (после сброса) встроенный (или внешний) процессор должен заполнить таблицу маршрутизации, регистр идентификации терминальных узлов и регистры адаптивной групповой маршрутизации начальными значениями (начальные значения для строк таблицы маршрутизации не определены, начальные значения для регистра идентификации терминальных узлов и регистров адаптивной групповой маршрутизации – '0'). В дальнейшем, в ходе работы МСК-01 возможна запись новых настроек в таблицу маршрутизации и в регистры адаптивной групповой маршрутизации.

В начале работы (после сброса) порты SpaceWire продолжают оставаться в состоянии сброса. Прежде, чем разрешить работу портов SpaceWire, необходимо записать в регистры скорости передачи для этих портов (TX_SPEED) значения, соответствующие частоте передачи 10 МГц (в соответствии со стандартом SpaceWire). После этого в регистры режима работы портов SpaceWire надо записать необходимые настройки (см. регистр MODE_CR).

При установке соединения по каналу SpaceWire происходит установка прерывания IRQ0 (см. Описание логики работы прерываний в МСК-01). В регистре CUR_CONNECTED отображается информация о том, установлено ли соединение по каждому из каналов SpaceWire. Если разряд i этого регистра установлен в '1', то по каналу i в данный момент времени установлено соединение.

Если в канале SpaceWire происходит ошибка, то устанавливается прерывание IRQ1 (см. Описание логики работы прерываний в МСК-01). В регистре CUR_ERRORED отображается информация о том, в каких каналах на данный момент времени соединение разорвано по причине ошибки в канале. Если разряд i этого регистра установлен в '1', то соединение в этом канале разорвано в результате ошибки. Если соединение по каналу не было установлено по причине того, что не было соответствующей команды от процессора или канал по инициативе процессора переведен в состояние сброса, то для этого канала соответствующий бит в регистре CUR_CONNECTED и CUR_ERRORED установлен в '0'.

Для того чтобы отправлять пакеты из конфигурационного порта в сеть, их необходимо записать в ОЗУ пакетов, после этого настроить DMA конфигурационного порта на передачу данных (см. Регистры DMA). Прежде чем отправлять в сеть пакет через конфигурационный порт, необходимо убедиться, что по всем каналам, по которым должен быть разослан данный пакет, установлено соединение. Если по каналам не установлено соединение, то пакет будет прочитан из памяти пакетов и отброшен.

Для того чтобы принимать пакеты из сети в конфигурационный порт, необходимо настроить DMA конфигурационного порта на прием данных (см. Регистры DMA). Если из сети приходит пакет, адресованный конфигурационному порту (порту 0), и DMA не настроен на прием данных (закончилась область данных и/или дескрипторов), то такой пакет не будет принят до тех пор, пока DMA не будет настроен на прием. (Таймаута, по истечении которого пакет мог бы быть отброшен, не предусмотрено).

Процессор может в любой момент прочитать текущее системное время из программно доступного регистра CUR_TIME. Процессор может в любой момент прочитать информацию о прохождении через МСК-01 распределенных прерываний и poll кодов из регистров ISR_H и ISR_L. При приходе из сети очередного корректного кода времени, распределенного прерывания или poll кода устанавливается прерывание IRQ2 (см. Описание логики работы прерываний в МСК-01).

Процессор конфигурационного порта может отправлять в сеть управляющие коды времени, распределенных прерываний и роII коды. Для этого необходимо записать значение соответствующего управляющего кода в регистр CONTROL_OUT.

4. ФУНКЦИОНАЛЬНОЕ ОПИСАНИЕ

4.1 Порт SpaceWire

В каждом порте SpaceWire реализованы:

- Аппаратное детектирование ошибок связи: рассоединение, ошибки четности.
- Встроенные LVDS приемопередатчики стандарта ANSI/TIA/EIA-644(LVDS).
- Встроенные в приемник LVDS резисторы-терминаторы.

Структурная схема порта SpaceWire приведена на Рисунок 4.1.

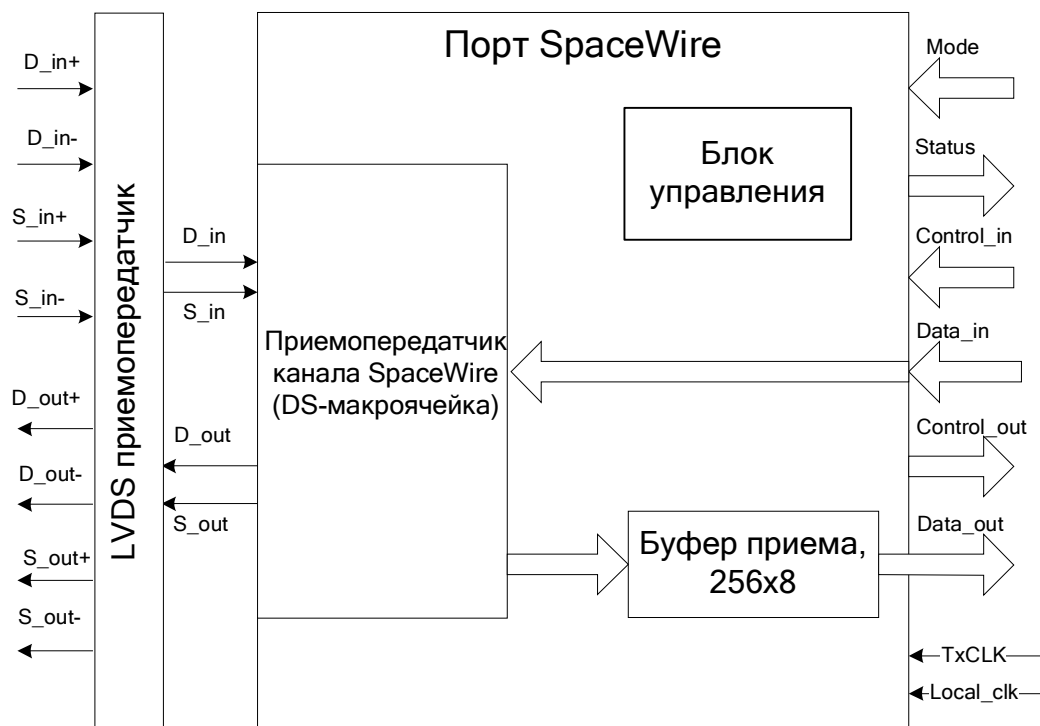


Рисунок 4.1 Структурная схема порта SpaceWire

Порт SpaceWire обеспечивает DS-кодирование и DS-декодирование данных и управляющих кодов при их передаче и приеме из канала SpaceWire. DS-кодирование выполняется при поступлении символов данных и концов пакетов из блока неблокирующего кросс-коммутатора или управляющих кодов от контроллера распределенных прерываний или контроллера распределения кодов времени. В результате в канал выдается последовательный поток бит на заданной блоком управления частоте.

При приеме из канала последовательного потока данных DS-декодирование позволяет выделить 8-разрядные символы данных и символы конца пакетов, а также управляющие коды. Символы данных и символы конца пакетов через буфер приема поступают в неблокирующий кросс-коммутатор. Управляющие коды поступают в контроллер распределенных прерываний или контроллер распределения кодов времени.

LVDS-приемопередатчик формирует LVDS-сигналы в соответствии со стандартом ANSI/TIA/EIA-644 при передаче последовательного потока бит в канал SpaceWire, а также осуществляет обратное преобразование при приеме дифференциальных сигналов из канала SpaceWire.

4.2 Блок регистров

Блок регистров состоит из компонента ведомого устройства AMBA АНВ и компонентов регистров. Каждый регистр реализован в виде отдельного компонента. Такая схема позволяет легко масштабировать блок в зависимости от числа каналов, реализованных в данной версии коммутатора. В данной реализации блок регистров включает в себя 96 программно доступных регистров (доступны встроенному и внешнему процессору на чтение и запись). Встроенный процессор может осуществлять обращения к регистрам через коммуникационную систему АНВ, внешний процессор может осуществлять обращения к регистрам через асинхронный интерфейс внешней памяти.

В блоке регистров осуществляется формирование сигналов прерываний для встроенного и внешнего процессора.

4.3 Таблица маршрутизации

Таблица маршрутизации включает в себя:

- Блок двухпортовой памяти размером 256 32-разрядных слов,
- интерфейс ведомого устройства на AMBA АНВ,
- интерфейс с контроллером управления коммутацией.

Интерфейс ведомого устройства на AMBA АНВ включает в себя следующие сигналы:

HRESET – системный сигнал сброса;

HCLK – сигнал тактирования;

HSEL – выбор устройства;

HADDR – адрес;

HWRITE – направление обмена;

HTRANS – команда;

HREADY_o – выходной сигнал готовности;

HREADY_i – входной сигнал готовности;

HRESP – сигнал подтверждения;

HWDATA – данные для записи в память;

HRDATA – данные, читаемые из памяти.

Интерфейс с контроллером управления коммутацией включает в себя следующие сигналы:

– MRE – сигнал разрешения чтения (поскольку по этому порту осуществляется только чтение, данный сигнал может быть всегда установлен в '1', однако, для снижения энергопотребления, этот сигнал устанавливается в '1', только когда действительно выполняется операция чтения);

– MADDR – адрес строки в таблице маршрутизации;

– MDOUT – данные, читаемые из таблицы маршрутизации.

Через интерфейс ведомого устройства на AMBA АНВ таблица маршрутизации может быть прочитана и записана встроенным или внешним процессором.

Через интерфейс с контроллером управления коммутацией контроллеры приемных интерфейсов портов SpaceWire осуществляют чтение строк таблицы маршрутизации, соответствующих заголовкам пакетов.

4.4 Неблокирующий кросс-коммутатор

Структурная схема неблокирующего кросс-коммутатора представлена на Рисунок 4.2.

Описание интерфейса компонента:

Системные сигналы:

– reset – асинхронный сигнал сброса;

– Clk – сигнал тактирования.

Интерфейс с портами SpaceWire:

– data_in – символы данных и концов пакетов, поступающие от портов SpaceWire;

- empty_in – сигналы, указывающие, есть ли еще информация для передачи от портов SpaceWire ;
- RE_in – сигналы готовности принять данные от портов SpaceWire;
- data_out – символы данных и концов пакетов для портов SpaceWire;
- empty_out – сигналы, указывающие, есть ли еще информация для передачи в порты SpaceWire;
- RE_out – сигналы готовности, указывающие портам SpaceWire, что можно передавать информацию.

Интерфейс с таблицей маршрутизации:

- Maddr – адрес строки маршрутизации, которая должна быть прочитана;
- Mre – разрешение чтения;
- Mdata – строка, читаемая из таблицы маршрутизации.

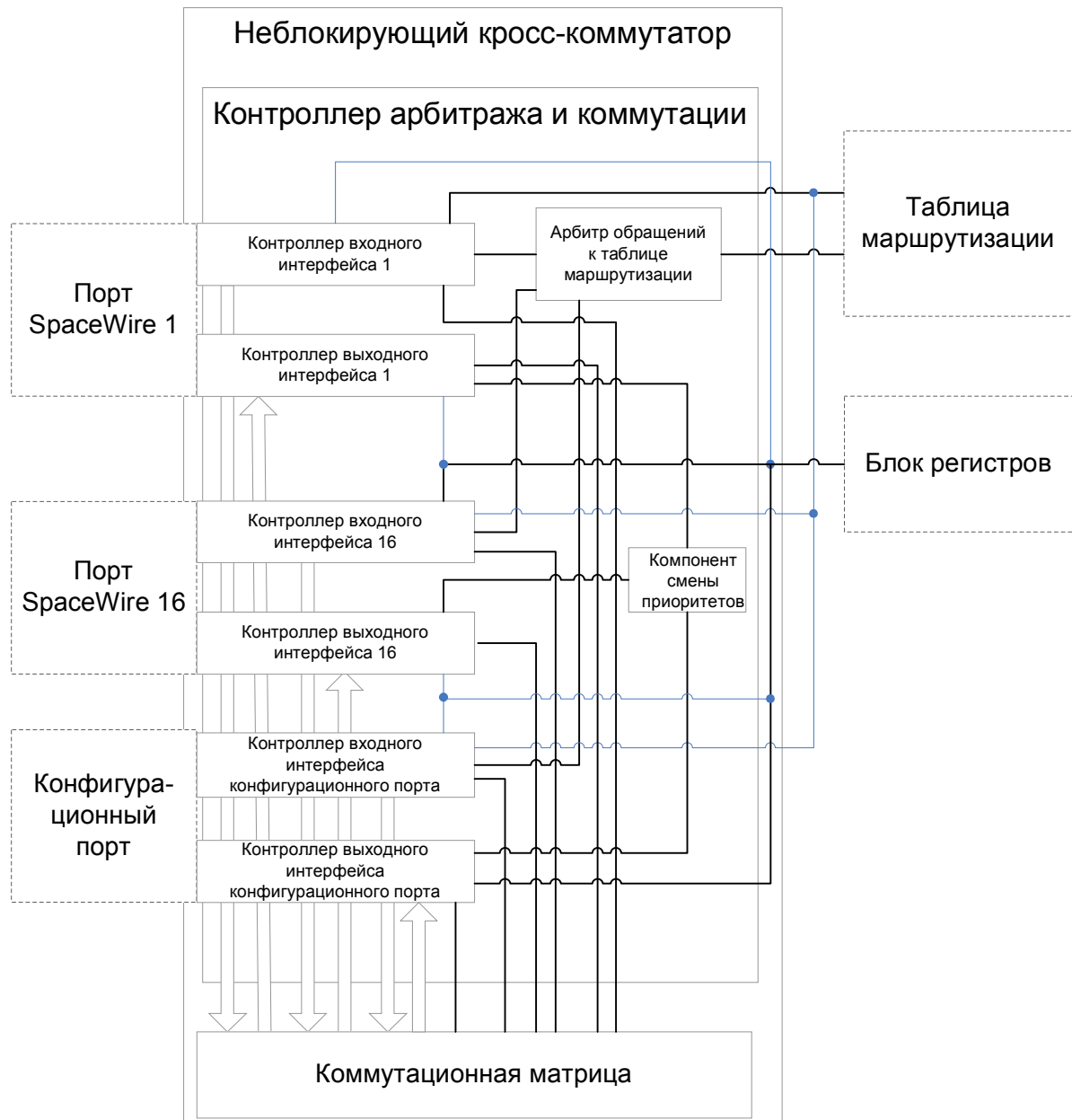


Рисунок 4.2 Неблокирующий кросс-коммутатор

Интерфейс с блоком регистров МСК-01:

- eq_regs – значения регистров адаптивной групповой маршрутизации;
- err_regs – значение регистра ошибок каналов ('1' в i разряде этого регистра соответствует отсутствию соединения по каналу с номером i);

sig_num – номер порта, который в данный момент времени имеет наивысший приоритет (поступает от вспомогательного компонента – компонента смены приоритетов), необходим для схемы арбитража с динамическими приоритетами. В состав неблокирующего кросс-коммутатора входят коммутационная матрица и контроллер арбитража и коммутации.

4.4.1 Коммутационная матрица

Коммутационная матрица включает в себя компоненты-каналы двух типов: первичные каналы и вторичные каналы (количество каналов каждого типа 17, что соответствует 16-ти портам SpaceWire и конфигурационному порту). Первичные каналы предназначены для передачи данных и сигналов действительности данных от приемных интерфейсов портов SpaceWire к передающим. Вторичные каналы предназначены для передачи сигналов разрешения чтения от передающих интерфейсов портов SpaceWire к приемным. Коммутационная матрица функционирует под управлением контроллера арбитража и коммутации. Для каждого первичного канала контроллер арбитража и коммутации определяет номер приемного интерфейса порта SpaceWire, который будет соединен с данным передающим интерфейсом порта SpaceWire, и сигнал действительности номера, указывающий, действительно ли в данный момент какой-либо интерфейс порта SpaceWire подключен к данному передающему интерфейсу. Если номер недействителен, то соответствующий выходной сигнал кросс-коммутатора empty_out устанавливается в '1'. Для каждого вторичного канала контроллер арбитража и коммутации определяет вектор разрядности 17 (соответственно 16 портов SpaceWire и конфигурационный порт). Если в *i*-ом разряде вектора '1', то сигнал готовности принять данные от порта SpaceWire с номером *i* должен учитываться при формировании общего сигнала готовности для данного порта SpaceWire. Это необходимо для обеспечения корректной рассылки данных от одного порта SpaceWire к нескольким.

4.4.2 Контроллер арбитража и коммутации

Контроллер арбитража и коммутации включает в себя контроллеры входных интерфейсов портов SpaceWire, арбитр обращений к таблице маршрутизации, контроллеры выходных интерфейсов каналов spaceWire, компонент смены приоритетов.

4.4.2.1 Контроллер входного интерфейса порта SpaceWire

Используется 16 таких компонентов, по одному для каждого порта SpaceWire. Этот компонент работает по следующему алгоритму. Если по каналу SpaceWire извне не поступают данные, контроллер входного интерфейса порта SpaceWire не выполняет каких-либо действий. Когда из канала SpaceWire поступает первое слово данных, не являющееся символом конца пакета, оно прочитывается и рассматривается как адрес данного пакета. (Следует отметить, что символы конца пакета в начале передачи сразу после установки соединения, также следующие друг за другом символы конца пакета считываются из порта SpaceWire и отбрасываются). Контроллер входного интерфейса порта SpaceWire прочитывает адрес пакета из порта SpaceWire, записывает его во внутренний регистр и выставляет его в качестве адреса обращения в таблицу маршрутизации. Параллельно он выставляет сигнал чтения из таблицы маршрутизации, который также поступает в арбитр обращений к таблице маршрутизации. Арбитр обращений к таблице маршрутизации определяет, какой из контроллеров входных интерфейсов портов SpaceWire в данный момент времени будет обращаться к таблице маршрутизации. После того, как из таблицы маршрутизации прочитана нужная строка, контроллер входного интерфейса порта SpaceWire определяет множество портов, в которые должен быть передан пакет, приоритет пакета и должен ли адрес пакета быть отброшен или передан дальше. Если прочитанная из таблицы маршрутизации строка оказалась пустой (в ней не указано ни одного порта назначения), то пакет прочитывается

из порта и отбрасывается. В противном случае определяется начальное множество портов, в которые будет передан пакет. Оно определяется с учетом регистров адаптивной групповой маршрутизации, регистра терминальных интерфейсов и текущего состояния портов. Для этого используется компонент выборки активного порта в группе. После того, как определено множество портов, которым должен быть передан пакет, контроллер входного интерфейса порта SpaceWire ожидает наступления фазы 0 установки связей (номер фазы определяется компонентом смены приоритетов и является одним и тем же для всех контроллеров входного интерфейса и выходного интерфейса). В фазе 0 контроллер входного интерфейса порта SpaceWire выставляет запрос контроллерам выходных интерфейсов порта SpaceWire в соответствии с выбранным множеством портов, в которые будет передан пакет. В следующей за этим фазе 1 контроллер входного интерфейса порта SpaceWire получает гранты от контроллеров выходных интерфейсов. Если гранты получены от всех запрашиваемых контроллеров выходных интерфейсов, в следующей фазе 2 контроллер входного интерфейса формирует для всех запрашиваемых контроллеров выходных интерфейсов сигналы подтверждения запроса, после чего начинается передача пакета. Если гранты получены не от всех контроллеров выходных интерфейсов, контроллер входного интерфейса не формирует для всех запрашиваемых контроллеров выходных интерфейсов сигналы подтверждения запроса (это означает, что он отказывается от использования этих портов). В этом случае контроллер входного порта повторно определяет множество портов, которым должен быть передан пакет. (Это необходимо вследствие того, что за время обращения занятые ранее порты, входящие в группы альтернативных портов могли освободиться, в портах, могли произойти события установки и разрыва соединения). Далее вновь выполняется попытка запроса множества выходных портов. Эти действия повторяются до тех пор, пока не будут получены гранты от всех затребованных портов. Поскольку во всех выходных интерфейсах портов SpaceWire используется единая схема приоритетов и фазы обмена для всех контроллеров определяются одинаково ситуация взаимоблокировок входных интерфейсов портов SpaceWire при запросах каждым из них нескольких выходных интерфейсов каналов SpaceWire исключена. После того, как контроллер входного интерфейса порта SpaceWire получил гранты на использование всех нужных ему выходных интерфейсов портов SpaceWire, происходит установка соединения – контроллер входного интерфейса порта SpaceWire выставляет сигнал управления для соответствующего ему вторичного канала. Контроллеры выходных интерфейсов портов SpaceWire, которые участвуют в обмене, формируют сигналы управления для соответствующих им первичных каналов. (Значения этих сигналов сохраняются неизменными до тех пор, пока не будет передан символ конца данного пакета.) После этого передается заголовок (лидирующий байт) пакета, если в соответствии с таблицей маршрутизации он должен быть передан дальше. Затем передаются остальные байты пакета. Каждый последующий байт прочитывается из порта SpaceWire после того, как предыдущий байт успешно передан во все порты, в которые рассылается данный пакет. После передачи символа конца пакета контроллер входного интерфейса порта SpaceWire прекращает соединение с контроллерами выходных интерфейсов портов SpaceWire и становится готовым к обработке следующего пакета.

4.4.2.2 Контроллер входного интерфейса конфигурационного порта

Данный контроллер отличается от остальных контроллеров входных интерфейсов портов SpaceWire тем, что не обращается к таблице маршрутизации для того, чтобы определить, куда должен быть отправлен пакет, а использует для этих целей первые четыре байта пакета (интерпретация их такая же, как в строке таблицы маршрутизации). Пакеты в контроллер входного интерфейса конфигурационного порта поступают из памяти пакетов. В эту память пакеты могут быть записаны встроенным или внешним процессором МСК-01. Из памяти пакеты прочитываются DMA конфигурационного

порта и через буфер передаются в контроллер входного интерфейса конфигурационного порта.

4.4.2.3 Контроллер выходного интерфейса порта SpaceWire

Контроллер осуществляет арбитраж обращений контроллеров входных интерфейсов портов SpaceWire. Для этого используется динамическая циклическая схема арбитража. Для определения входного интерфейса порта SpaceWire, имеющего наивысший приоритет в текущий момент времени, всеми контроллерами выходных интерфейсов портов SpaceWire используется один компонент смены приоритетов. Для определения тактов, в которых будут анализироваться запросы, выставляться гранты и анализироваться наличие подтверждения запроса, используется единая схема смены фаз установки связей. Синхронная смена приоритетов и фаз установки связей позволяет исключить взаимоблокировки между контроллерами входных интерфейсов портов SpaceWire.

Если в порте SpaceWire, соответствующем данному контроллеру выходного интерфейса, в текущий момент на физическом уровне соединение не установлено (порт не работает вследствие того, что для него не было дано команды на установку соединения или произошел разрыв соединения в результате ошибки в канале), то этот контроллер выставляет в ответ на все поступившие к нему запросы сигналы гранта. Благодаря этому отбрасываются пакеты, которые направлены в неработающие каналы, что необходимо для того, чтобы такие пакеты не заблокировали всю коммуникационную систему. Схема выбора выходных портов при наличии альтернативных каналов организована таким образом, чтобы если в группе альтернативных каналов присутствует хотя бы один канал, по которому в данный момент установлено соединение (соответствующий порт в рабочем состоянии), то для передачи будет выбираться именно он. Это позволяет исключить неоправданное отбрасывание пакетов.

Если по каналу, соответствующему данному контроллеру выходного интерфейса, в текущий момент установлено соединение и не осуществляется передача пакета, то он в фазе 0 установки связей по результатам арбитража выбирает контроллер входного порта, из которого может приниматься очередной пакет. В фазе 1 для этого контроллера выставляется грант. И если в фазе 2 поступает подтверждение запроса, то соединение считается установленным, в соответствии с этим выставляются сигналы управления для коммутационной матрицы, которые сохраняются на все время передачи пакета. Если же подтверждение запроса не поступило, то контроллер выходного порта в следующей фазе 0 установки связей вновь может выбрать контроллер входного порта.

4.4.2.4 Арбитр обращений к таблице маршрутизации

Этот блок предназначен для приема запросов на обращение к таблице маршрутизации от контроллеров входных интерфейсов портов SpaceWire. Он определяет, какой из контроллеров в данный момент будет обращаться к таблице.

4.4.2.5 Компонент смены приоритетов

Компонент смены приоритетов определяет номер порта SpaceWire, который в данный момент времени будет иметь наивысший приоритет. В начале работы схемы наивысший приоритет имеет SWPORT1, далее наивысший приоритет переходит к SWPORT 2 и т. д. Смена приоритетов осуществляется через фиксированное количество тактов. Данное количество тактов является программно настраиваемым. Этот компонент также выполняет функцию определения фазы установки связи между контроллерами входных и выходных интерфейсов портов SpaceWire. В фазе 0 контроллеры входных интерфейсов могут выставлять запросы, в фазе 1 контроллеры выходных интерфейсов могут выставлять гранты, в фазе 2 контроллеры входных интерфейсов могут выставлять подтверждения запросов (в случае получения грантов).

(Поскольку контроллер распределенных прерываний также использует динамическую циклическую смену приоритетов, выход данного компонента связан с соответствующим сигналом в интерфейсе контроллера арбитража и коммутации.)

4.5 Контроллер распределения кодов времени

Описание интерфейса компонента:

Системные сигналы: reset – асинхронный сигнал сброса;

Clk – сигнал тактирования.

Интерфейс с каналами SpaceWire:

control_in – значения управляющих кодов с выходов портов;

valid_in – сигналы, подтверждающие действительность управляющих кодов с выходов портов;

control_out – значения управляющих кодов для подачи на входы портов (на входы портов поступают после прохождения компонента арбитража управляющих кодов);

valid_out – значения, подтверждающие действительность управляющих кодов для подачи на входы портов (на входы портов поступают после прохождения компонента арбитража управляющих кодов);

WE – сигналы разрешения записи управляющих кодов в порты.

Интерфейс с блоком регистров МСК-01:

eq_regs – значения регистров адаптивной групповой маршрутизации;

err_regs – значение регистра ошибок каналов (1 в I разряде этого регистра соответствует отсутствию соединения по данному каналу);

out_time – значение для записи в регистр текущего времени (этот регистр дублирует значение базового регистра текущего времени во временном домене HCLK);

time_w – разрешение записи в регистр текущего времени;

base_eq – текущая выборка каналов в соответствии с регистрами адаптивной групповой маршрутизации.

Структурная схема контроллера распределения кодов времени представлена на Рисунок 4.3.

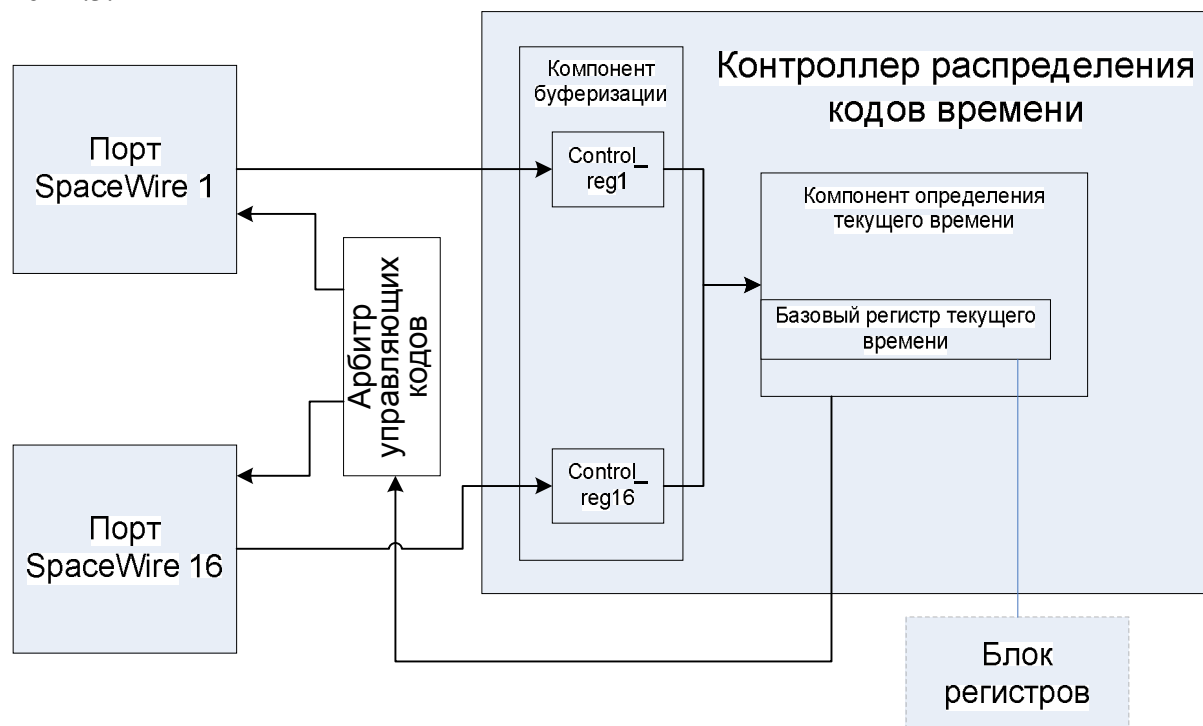


Рисунок 4.3 Структурная схема контроллера распределения кодов времени

Контроллер распределения меток времени включает в себя два компонента: компонент буферизации и компонент определения текущего времени.

4.5.1 Компонент буферизации

Управляющие коды могут поступать на выход канала SpaceWire каждые два такта системной частоты. За этот интервал времени значения управляющего кода времени должно быть записано, так как в противном случае оно может быть утрачено в результате приема другого управляющего кода. Компонент определения текущего времени может обработать не более одного кода времени за один такт. Для того чтобы не происходило потерь кодов времени, пришедших через короткие интервалы времени по различным каналам SpaceWire, используется компонент буферизации.

Компонент буферизации содержит 17 буферов (по количеству портов SpaceWire плюс конфигурационный порт – процессор конфигурационного порта может отправлять в сеть коды времени, записывая их в регистр кода времени конфигурационного порта). Если на вход буфера поступает управляющий код времени, то он записывается в буфер. Буфер выставляет значение кода времени и запрос на его обработку для компонента определения текущего времени.

4.5.2 Компонент определения текущего времени

Этот компонент работает по следующей схеме. Каждый такт проверяется, имеются ли запросы на обработку кодов времени от компонента буферизации. Если имеется запрос, то прочитывается значение кода времени. При арбитраже запросов от буферов используется алгоритм с абсолютными приоритетами (чем меньше номер канала, тем выше его приоритет). В силу особенностей потока входных кодов времени схема арбитража при нормальной работе не влияет на поток выходных кодов времени. (В общем случае коды времени поступают не часто и коды времени, меньшие, чем значение базового регистра текущего времени, возникают крайне редко.) Данная схема арбитража выбрана в силу того, что она реализуется с наименьшими аппаратными затратами.

Далее прочитанное значение кода времени сравнивается со значением в базовом регистре текущего времени и выполняется его обработка в соответствии со стандартом SpaceWire.

Если значения совпадают, то не выполняется никаких действий.

Если принятое значение на «1» превосходит текущее значение или текущее значение '63', а принятое – '0', то в базовый регистр текущего времени записывается новое значение. Это значение выдается во все каналы SpaceWire с учетом значений регистров адаптивной групповой маршрутизации и регистров ошибок каналов: значение не выдается в канал, из которого оно было принято и во все каналы, принадлежащие той же группе, далее значение выдается строго в один и только один из каналов каждой группы. Для выборки каналов используется вспомогательный компонент, выбирающий по одному каналу из каждой группы (этот компонент вынесен за пределы контроллера распределения кодов времени, и алгоритм работы этого компонента описан отдельно, так как он используется также для контроллера распределенных прерываний и неблокирующего кросс-коммутатора). Значение кода времени держится на входе каждого из портов до тех пор, пока оно не будет принято портом либо порт не перейдет в нерабочее состояние (в результате разрыва соединения или истечения времени, указанного программно настраиваемом регистре таймаута). Передача следующего кода времени начинается только после того, как предыдущий код времени был выдан во все порты, в которые было запланировано его передать.

Если принятое значение меньше значения в базовом регистре текущего времени или более чем на '1' больше, чем в базовом регистре текущего времени, то оно записывается в базовый регистр текущего времени, однако, на входы портов SpaceWire не подается.

4.6 Контроллер распределенных прерываний

Описание интерфейса компонента:

системные сигналы: `reset` – асинхронный сигнал сброса;

`clk` – системный сигнал тактирования.

Интерфейс с портами SpaceWire: `control_in` – значения управляющих кодов с выходов портов SpaceWire;

`valid_in` – значения сигналов подтверждения с выходов подтверждения управляющих кодов портов SpaceWire;

`control_out` – значения управляющих кодов, подаваемые на входы портов SpaceWire (на входы портов поступают после прохождения компонента арбитража управляющих кодов);

`valid_out` – значения действительности управляющих кодов, подаваемых на входы портов SpaceWire;

`WE` – сигналы готовности от портов SpaceWire.

Интерфейс с блоком регистров МСК-01:

`eq_regs` – значения регистров адаптивной групповой маршрутизации;

`err_regs` – значение регистра ошибок каналов («1» в *i*-ом разряде этого регистра соответствует отсутствию соединения по данному каналу);

`base_eq` – текущая выборка каналов в соответствии с регистрами адаптивной групповой маршрутизации;

`ISR_out` – значение для записи в регистр ISR (этот регистр дублирует значение базового регистра текущего времени во временном домене HCLK);

`ISR_w` – разрешение записи в регистр ISR;

`cur_num` – номер порта, который в данный момент времени имеет наивысший приоритет (поступает от вспомогательного компонента – компонента смены приоритетов; этот компонент вынесен за пределы контроллера распределенных прерываний, поскольку используется также для схемы арбитража в неблокирующем кросс-коммутаторе), необходим для схемы арбитража с динамическими приоритетами.

Структурная схема контроллера распределенных прерываний представлена на Рисунок 4.4.

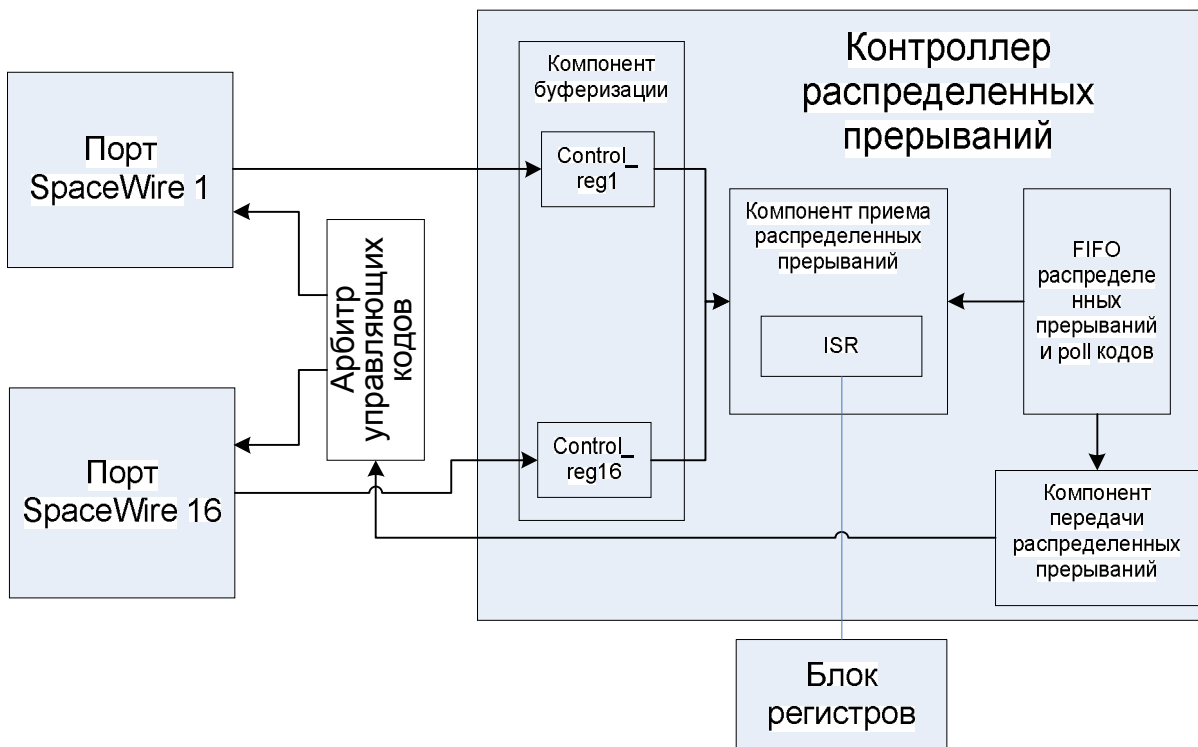


Рисунок 4.4 Структурная схема контроллера распределенных прерываний

Контроллер распределенных прерываний содержит следующие компоненты: компонент буферизации, компонент приема распределенных прерываний и poll кодов, FIFO распределенных прерываний и poll кодов и компонент передачи распределенных прерываний и poll кодов.

4.6.1 Компонент буферизации

Структура и логика работы этого компонента буферизации аналогична используемой в контроллере распределения кодов времени. Отличие в том, что в буферах защелкиваются управляющие коды, если они являются распределенными прерываниями или poll кодами. (Процессор конфигурационного порта может быть источником кодов распределенных прерываний и poll кодов).

4.6.2 Компонент приема распределенных прерываний

Этот компонент работает следующим образом. Каждый такт проверяется, имеются ли запросы от буферов. Если имеется запрос, то прочитывается значение кода распределенного прерывания или poll кода. При арбитраже запросов используется алгоритм с динамическими циклическими приоритетами. Его использование гарантирует, что запрос от любого буфера будет обработан за конечное время.

Далее если прочитано распределенное прерывание и в соответствующем разряде регистра ISR стоит '0' или прочитан poll код и в соответствующем разряде регистра ISR стоит '1', то значение управляющего кода и номер порта, из которого он поступил, записывается в буфер. В противном случае не выполняется никаких действий.

4.6.3 FIFO распределенных прерываний и poll кодов

Используется стандартный компонент – синхронный буфер – вход и выход буфера работают на одной и той же системной частоте. Длина буфера 64 слова определяется максимальным количеством распределенных прерываний и poll кодов, одновременно находящихся в системе. Разрядность слова 13. Разряды (0...7) содержат значение управляющего кода, Разряды (8...13) – номер порта, являющегося источником данного управляющего кода.

4.6.4 Компонент передачи распределенных прерываний

Если буфер не пуст, то из него прочитывается очередное слово. На основе номера порта источника данного управляющего кода (содержится в слове, прочитанном из буфера), значений регистров адаптивной групповой маршрутизации и регистра ошибок каналов определяется множество портов, в которые будет передан данный управляющий код. (Алгоритм выбора аналогичен осуществляемому в контроллере распределения кодов времени.). Далее управляющий код рассылается во все выбранные порты. Только после того, как он разослан, может быть выбрано следующее слово из буфера. Если скорость передачи по каналам отличается очень существенно, это может привести к некоторому снижению скорости распространения распределенных прерываний и roll кодов по сети. Однако передача следующего управляющего кода до окончания передачи предыдущего приводит к очень сильному усложнению схемы. Кроме того, если распределенные прерывания и roll коды обрабатываются очень быстро, возможно возникновение ситуации, когда управляющий код с номером i еще не отправлен в медленный канал, однако код с таким же номером уже вновь поступил в контроллер распределенных прерываний. Это может привести к некорректному поведению системы в целом.

4.7 Компонент арбитража управляющих кодов

Этот компонент получает запросы на передачу управляющих кодов от компонента распределения кодов времени и компонента обработки распределенных прерываний и передает управляющие коды на входы портов SpaceWire. Используется дисциплина арбитража с абсолютными приоритетами. Компонент распределения кодов времени имеет наиболее высокий приоритет. Арбитраж выполняется для каждого порта SpaceWire отдельно. Используемая дисциплина арбитража позволяет исключить возможные ситуации взаимоблокировок портов.

4.8 Компонент выборки активного канала в группе

Компонент выборки активного канала (порта SpaceWire) в группе работает по следующему алгоритму.

Выполняется последовательный просмотр регистров адаптивной групповой маршрутизации. Для каждой группы определяется активный канал с учетом текущих приоритетов и состояния каналов. Среди входящих в группу каналов выбирается работоспособный канал (т.е. в нем на физическом уровне установлено соединение), который в данный момент имеет наивысший приоритет. При этом учитывается занятость каналов: если в группе имеются свободные каналы, то выбор осуществляется только среди них. Время работы схемы в зависимости от технологии реализации может занимать различное количество тактов. Соответственно это определяет частоту обновления текущей выборки каналов в соответствии с регистрами адаптивной групповой маршрутизации.

4.9 ОЗУ пакетов

Память пакетов включает в себя:

- два блока памяти размером 4К 32-х разрядных слов,
- интерфейс ведомого устройства на AMBA AHB,
- интерфейс с DMA.

4.10 Блок DMA конфигурационного порта

DMA содержит четыре блока для работы с парой каналов на запись в память, и парой – на чтение из памяти. Данные, как на прием, так и на передачу имеют формат 32-разрядного слова, содержание которого прозрачно для DMA. При работе с SWPORT DMA осуществляет обмен данными и дескрипторами с памятью. Поэтому в названиях

сигналов присутствуют фрагменты <DATA> (для каналов, работающих с данными), и <DESC> (для каналов, работающих с дескрипторами). Указанное не относится к сигналу REG_DATA.

DMA содержит интерфейс с памятью, с которой производит обмен данными из указанных каналов. Доступ каналов к памяти осуществляется по приоритетному принципу, при этом приоритеты доступа меняются динамически в ходе работы DMA. DMA содержит специальный регистр размера максимальной транзакции, ограничивающий транзакции с памятью указанной величиной.

DMA содержит набор управляющих регистров, с помощью которых можно настроить адреса и размеры областей памяти для каждого канала, запретить или разрешить работу каналов, получить информацию о состоянии работы DMA в текущий момент времени. DMA содержит вывод прерывания, который сообщает о том, что один из каналов DMA требует перенастройки.

4.11 Блок CSR

Блок CSR входит в состав RISC-ядра и содержит три 32-разрядных регистра:

- регистр управления и состояния CSR (не содержит изменений в связи с использование RISC-ядра в MCK-01);
- регистр запросов прерывания QSTR;
- регистр маски MASKR (не содержит изменений в связи с использование RISC-ядра в MCK-01).

Формат регистра QSTR приведен в Таблица 4.1. Регистр прерываний доступен только по чтению.

Таблица 4.1 Формат регистра запросов прерывания QSTR

Номер разряда	Условное обозначение прерывания	Название прерывания
0	IRQ0	Прерывание IRQ0 от коммутатора уровня 2
1	IRQ1	Прерывание IRQ1 от коммутатора уровня 2
2	IRQ2	Прерывание IRQ2 от коммутатора уровня 2
3	IRQ3	Прерывание IRQ3 от коммутатора уровня 2
4	Uart	Прерывание от UART
18...5	-	Резерв (установлены в 0)
19	Compare	Прерывание от таймера CPU
31...20	-	Резерв (установлены в 0)

Исходное состояние регистра QSTR – нули.

4.12 MBA (контроллер MPORT)

Контроллер MPORT представляет собой мост интерфейса асинхронной статической памяти с шиной АНВ. Со стороны интерфейса асинхронной памяти модуль является ведомым (эмулирует память), со стороны интерфейса АНВ – ведущим. Модуль не включает в себя тристабильные буферы шины данных.

Группа сигналов CSn, nRD, nWE поступает в MBA из другого временного домена. Для каждого из них используется схема перехода через временной домен, включающая в себя два триггера, функционирующих на частоте HCLK.

Запрос HBUSREQ формируется комбинаторно, инверсией сигнала CSn. Далее после получения сигнала HGRANT анализируются сигналы RDn и WEn.

По обнаружению активного состояния сигналов WEn или RDn мост начинает соответствующую транзакцию АНВ (Рисунок 4.5, Рисунок 4.6).

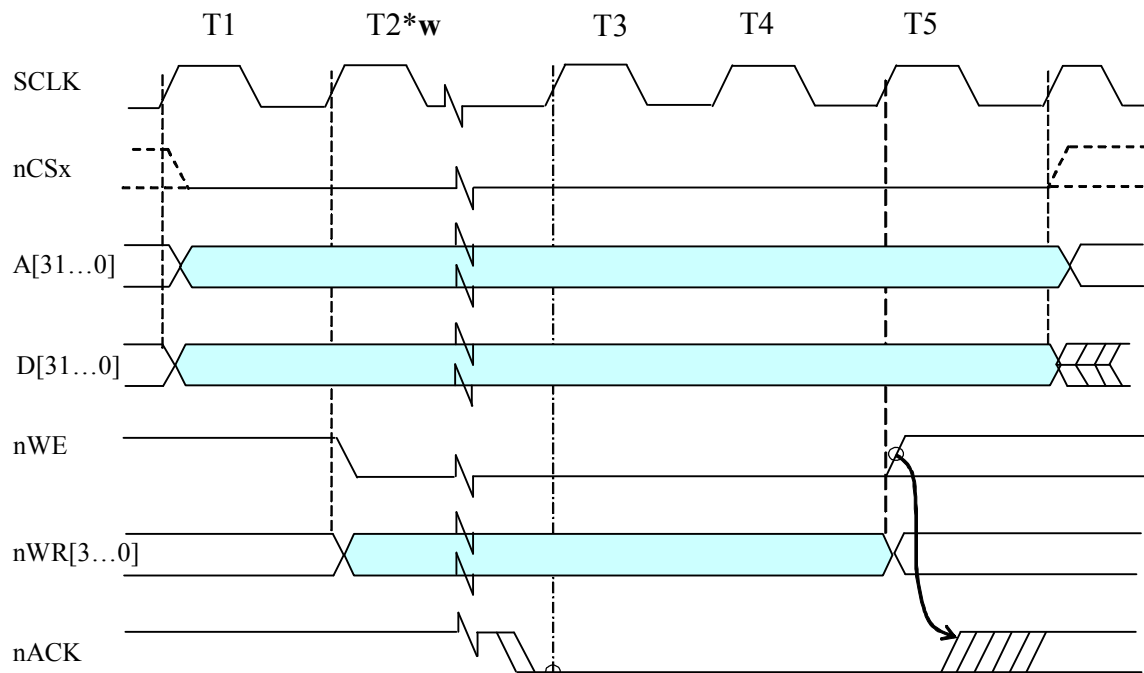


Рисунок 4.5 Запись данных

Получив подтверждение (в виде $HREADY=1$, так как $HRESP$ игнорируется: не существует механизма передачи информации о незавершенной транзакции процессору через реализуемый интерфейс, поэтому АНВ обязана гарантировать доставку информации, доступна лишь установка тактов ожидания через $HREADY=0$), мост устанавливает сигнал $nACK$ и ожидает деактивации сигналов Rd_n и WEn . После этого деактивируется $nACK$, и мост готов к следующей транзакции.

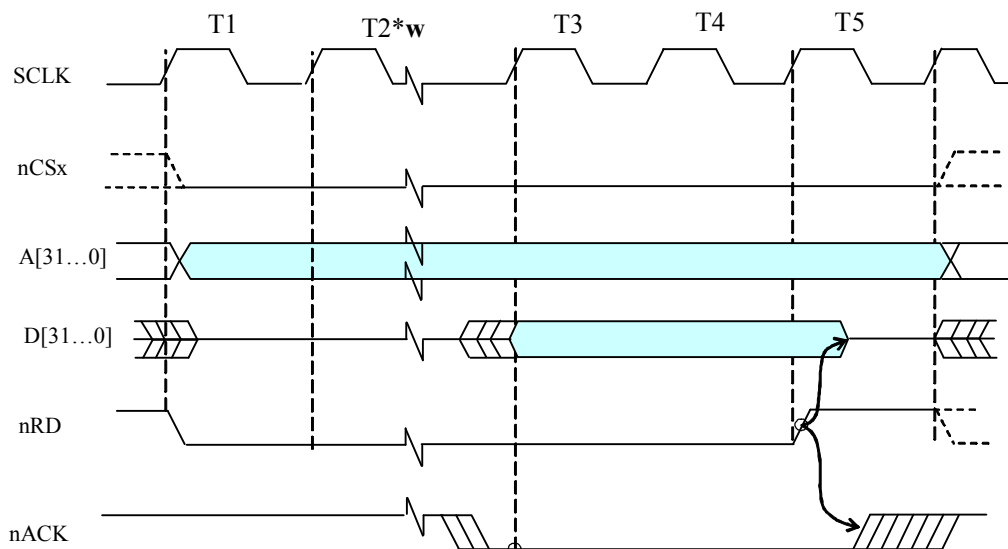


Рисунок 4.6 Чтение данных

4.13 Блок коммуникационной системы АНВ

К блоку коммуникационной системы подключены два ведущих устройства: контроллер MPORT и мост АНВ'/АНВ; и 4 ведомых устройства – таблица маршрутизации, память пакетов, блок регистров МСК-01, блок регистров DMA. Блок коммуникационной системы реализован как неблокирующий коммутатор. Компоненты арбитража функционируют по схеме с абсолютными приоритетами: контроллеру MPORT соответствует приоритет 0 (наивысший), мосту АНВ'/АНВ – приоритет 1.

4.14 Мост АНВ'/АНВ

Мост АНВ'/АНВ предназначен для упрощения и структуризации интерфейса между уровнем 2 и уровнем 3 МСК-01. На шине АНВ' он представлен интерфейсом ведомого устройства, в коммуникационной системе АНВ он представлен интерфейсом ведущего устройства. Данный мост обеспечивает доступ на чтение и запись к памяти пакетов, таблице маршрутизации, регистрам МСК-01 и регистрам DMA со стороны внутреннего процессора МСК-01, расположенного на уровне 3. Вследствие того, что контроллер MPORT, также подключенный к коммуникационной системе АНВ, поддерживает протокол обмена по внешней шине памяти без сигнала подтверждения, необходимо, чтобы доступ к ведомым устройствам на АНВ гарантированно предоставлялся ему за фиксированное количество тактов. Мост АНВ'/АНВ все транзакции преобразует в однословные последовательные транзакции. Это обеспечивает завершение транзакции за один такт.

5. ИНФОРМАЦИЯ О ПРИМЕНЕНИИ

МСК-01 может функционировать под управлением внутреннего процессора, внешнего процессора или при совместном управлении внутреннего и внешнего процессора. Внутренний и внешний процессор могут выполнять так же функции терминальных узлов.

Если МСК-01 функционирует под управлением только внутреннего процессора (Рисунок 5.1), к его интерфейсу MPORT должно быть подключено ПЗУ, содержащее программу функционирования внутреннего процессора. Кроме того, в дополнение к внутренней памяти МСК-01, к этому интерфейсу могут быть подключены дополнительные ОЗУ (RAM, SDRAM). Это может быть актуально, если внутренний процессор МСК-01 выполняет так же функции терминального узла.

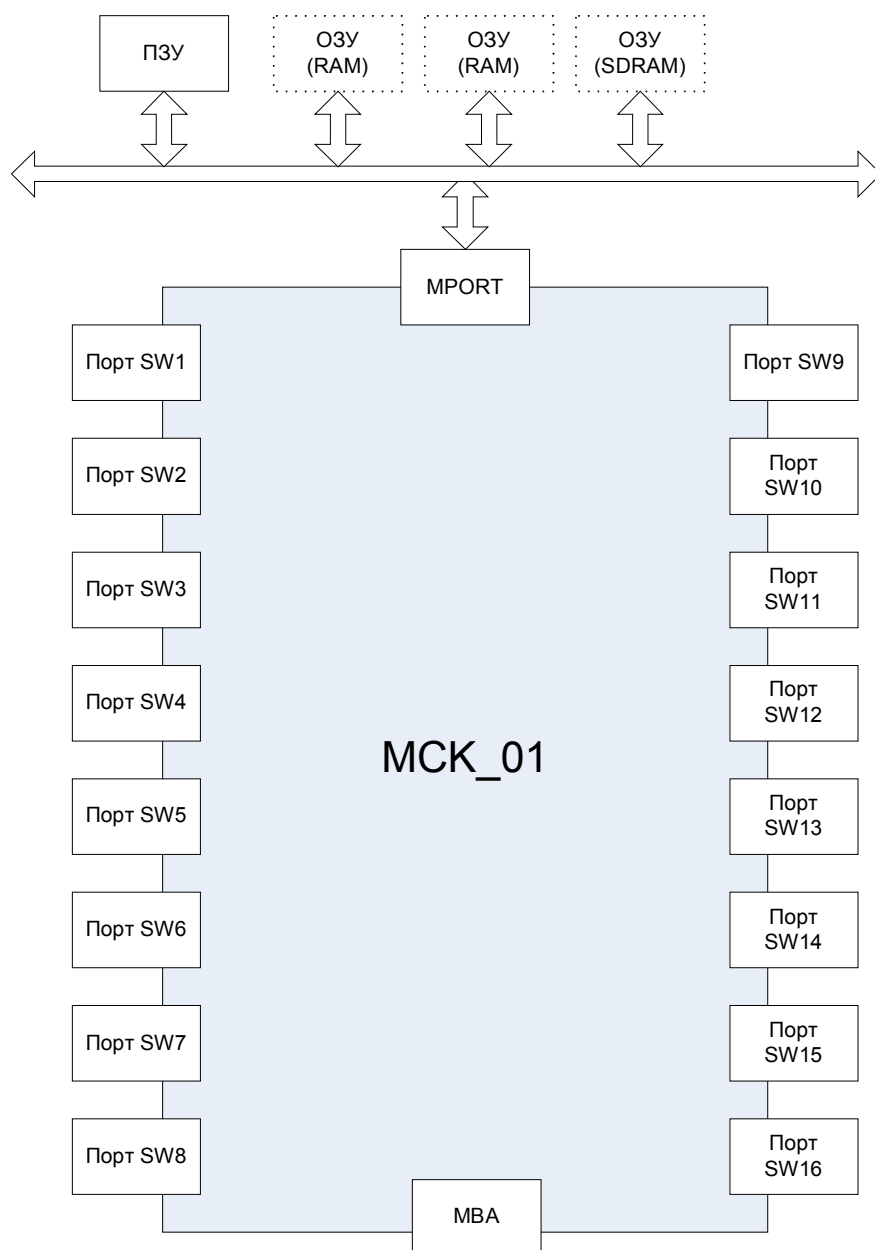


Рисунок 5.1 Структурная схема используемая при функционировании МСК-01 только под управлением внутреннего процессора

Если МСК-01 функционирует под управлением только внешнего процессора (Рисунок 5.2), к его интерфейсу МВА должен быть подключен внешний процессор. (На ту же шину памяти могут быть подключены ПЗУ и ОЗУ, необходимые для функционирования внешнего процессора). Внешний процессор так же может функционировать в режиме терминального узла.

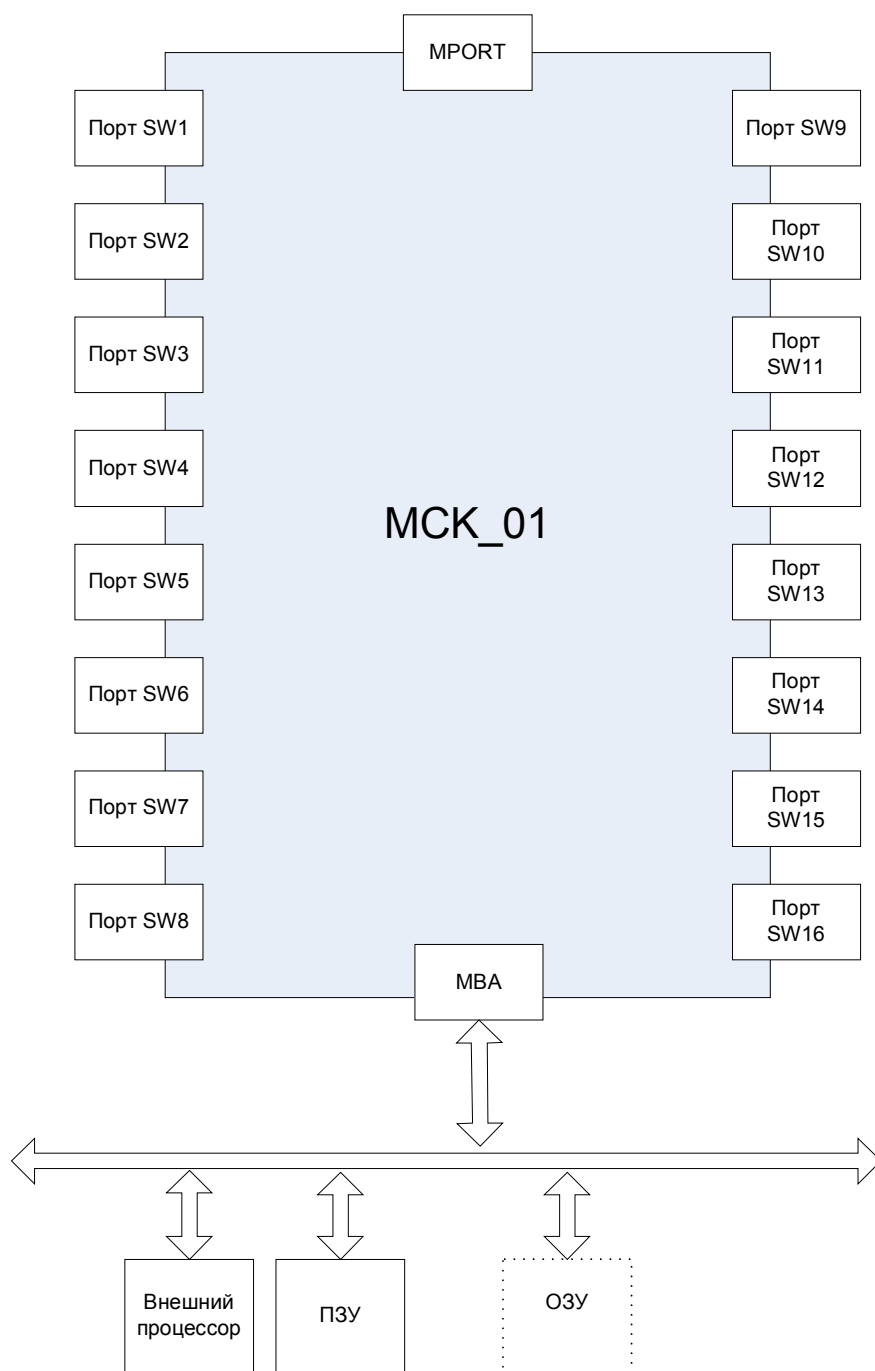


Рисунок 5.2 Структурная схема используемая при функционировании МСК-01 только под управлением внешнего процессора

МСК-01 может функционировать под управлением внутреннего и внешнего процессора (Рисунок 5.3). В этом случае согласование действий процессоров выполняется программно. Для этого может быть использована внутренняя память МСК-01 (ОЗУ пакетов), а так же разряды 31..5 регистра состояния. Каждый из процессоров может выполнять функции терминального узла.

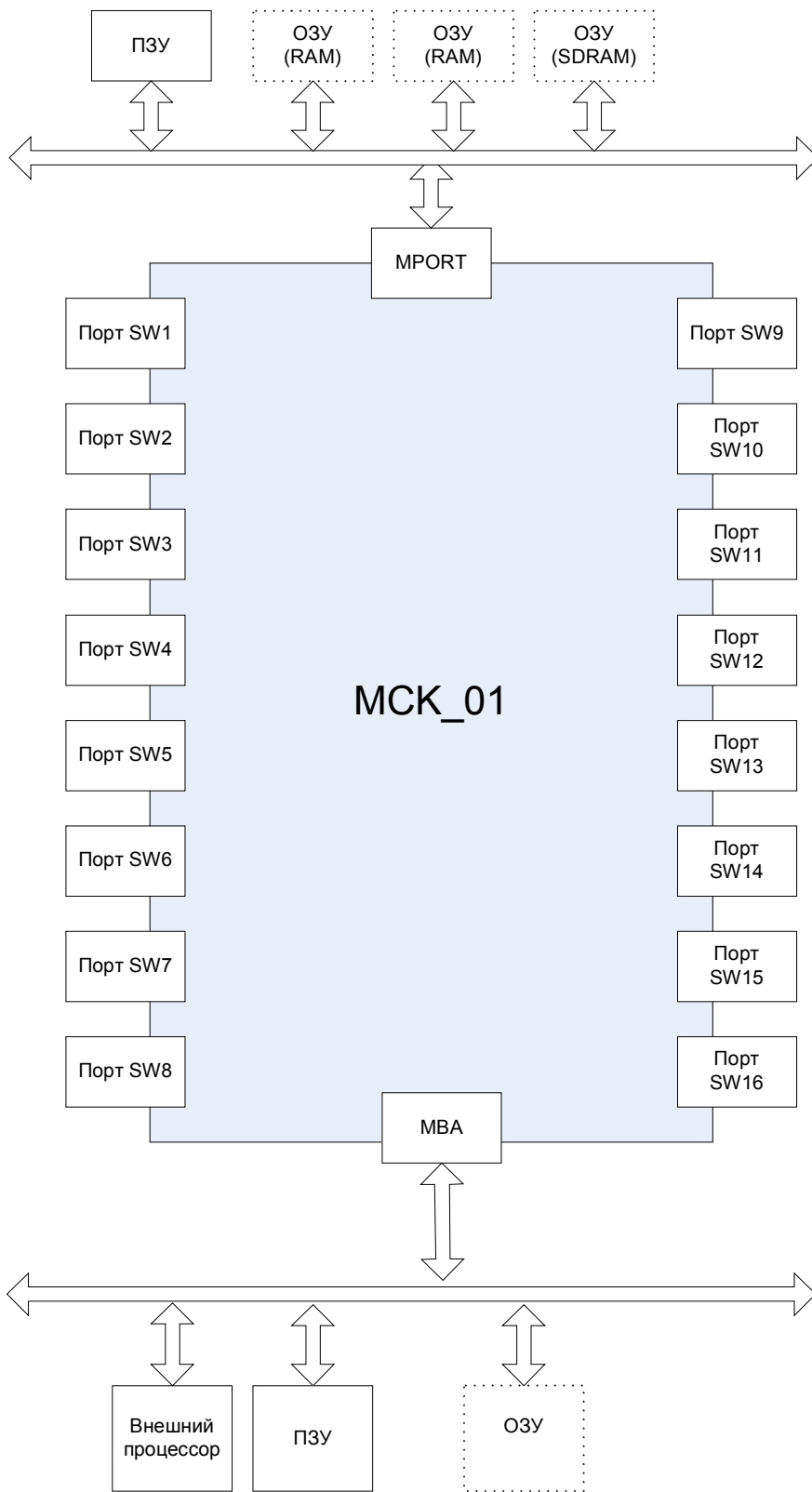


Рисунок 5.3 Структурная схема используемая при функционировании МСК-01 под управлением внутреннего и внешнего процессора

В этом разделе содержится информация об использовании блоков в реальных СнК. Представляется история создания блока, а также имевшиеся изменения в реализации блока.

6. ЭЛЕКТРИЧЕСКИЕ ПАРАМЕТРЫ

6.1 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 $\pm 5\%$.

Напряжения электропитания U_{CC1} и U_{CC2} необходимо подавать и снимать одновременно с разбросом не более 1 мс. Фронт нарастания напряжений электропитания должен быть не более 1 мс.

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 μ 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.

6.2 Электрические параметры

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 CVDD, мА при $U_{CC1}=3,47$ В, $U_{CC2}=2,63$ В, ХТИ=0	I_{CC2}	-	30
Ток потребления статический по цепи PVDD, мА при $U_{CC1}=3,47$ В, $U_{CC2}=2,63$ В, ХТИ=0	I_{CC1}	-	30
Ток потребления динамический по цепи CVDD, мА, при $U_{CC1}= 3,47$ В, $U_{CC2}= 2,63$ В и рабочей частоте ядра микросхемы 80 МГц	I_{OCC2}	-	500
Ток потребления динамический по цепи PVDD, мА, при $U_{CC1}= 3,47$ В, $U_{CC2}= 2,63$ В и рабочей частоте ядра микросхемы 80 МГц	I_{OCC1}	-	150
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_1	-	20

Наименование параметров, единица измерения, режим измерения	Буквенное обозначение	Норма	
		не менее	не более
Выходное дифференциальное напряжение по LVDS выходам, мВ	U_{OD}	300	400
Входное дифференциальное сопротивление по LVDS входам, Ом	R_{ID}	90	120

6.3 Пределно-допустимые и предельные электрические режимы эксплуатации

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

Таблица 6.2. Значения пределно-допустимых и предельных электрических режимов эксплуатации

Наименование параметра	Буквенное обозначение	Норма			
		Пределно 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

7. ОПИСАНИЕ ВНЕШНИХ ВЫВОДОВ

МСК-01 имеет следующие группы выводов:

- порт сопряжения с внешней памятью;
- порт сопряжения с внешним процессором;
- порты SpaceWire;
- системные сигналы;
- порт JTAG;
- UART;
- электропитание.

Назначение выводов МСК-01 приведено в Таблица 7.1 - Таблица 7.7.

Таблица 7.1. Порт сопряжения с внешней памятью

Название Вывода	Количество	Тип	Назначение
A[27:0]	28	O	Шина адреса.
D[31:0]	32	IO	Шина данных
nWE	1	O	Запись асинхронной памяти
nRD	1	O	Чтение асинхронной памяти
nACK	1	I	Готовность асинхронной памяти
nCS[3:0]	4	O	Разрешение выборки блоков внешней памяти
SRAS	1	O	Строб адреса строки
SCAS	1	O	Строб адреса колонки
SWE	1	O	Разрешение записи
DQM[3:0]	4	O	Маска выборки байта
SCLK	1	O	Тактовая частота работы
A10	1	O	10 разряд адреса
BA[1:0]	2	O	Номер банка
Всего 78 выводов			

Таблица 7.2. Порт сопряжения с внешним процессором

Название Вывода	Количество	Тип	Назначение
AS[15:0]	16	I	Шина адреса.
DS[31:0]	32	IO	Шина данных
nWES	1	I	Сигнал записи
nRDS	1	I	Сигнал чтения
nACKS	1	O	Сигнал готовности
ACKS	1	O	Сигнал готовности
nCSS	1	I	Сигнал разрешения выборки МСК-01
Всего 53 вывода			

Таблица 7.3. Порт JTAG

Название Вывода	Количество	Тип	Назначение
TCK	1	I	Тестовый тактовый сигнал (JTAG)
TRST	1	I	Установка исходного состояния (JTAG)
TMS	1	I	Выбор режима теста (JTAG)
TDI	1	I	Вход данных теста (JTAG)
TDO	1	O	Выход данных теста (JTAG)
Всего 5 выводов			

Таблица 7.4. Порты SpaceWire

Название вывода	Количество	Тип	Назначение
Боевые			
DINP[15:0]	16	I	Вход данных положительный
DINN[15:0]	16	I	Вход данных отрицательный
SINP[15:0]	16	I	Вход строба положительный
SINN[15:0]	16	I	Вход строба отрицательный
DOUTP[15:0]	16	O	Выход данных положительный
DOUTN[15:0]	16	O	Выход данных отрицательный
SOUTP[15:0]	16	O	Выход строба положительный
SOUTN[15:0]	16	O	Выход строба отрицательный
Всего 128 выводов			

Таблица 7.5. Системные выводы

Название вывода	Количество	Тип	Назначение
nIRQ[3:0]	4	I	Запрос прерывания
BYTE	1	I	Разрядность шины данных 6 банка внешней памяти: 0 – 32 разряда; 1 – 8 разрядов.
PLL_EN	1	I	Разрешение работы PLL: 0 – системная тактовая частота коммутатора, а также частота передачи портов SpaceWire равны входной частоте ХТ1 1 - системная тактовая частота коммутатора, а также частота передачи портов SpaceWire поступает из соответствующей PLL.
ХТ110, ХТ010	2	I, O	Выводы для подключения внешнего кварцевого резонатора частотой 10 МГц. Из нее получается 2 МГц для PLL_TX. На вывод ХТ110 можно подать частоту от внешнего генератора, при этом вывод ХТ010 должен быть незадействованным.
ХТ1	1	I	Вход тактовой частоты 10 МГц. Эта частота поступает на умножитель частоты. С умножителя частота поступает на тактирование цифровой части коммутатора. Вход ХТ1, как правило, соединяется с входом ХТ110
nRST	1	I	Сигнал установки исходного состояния
nRSTM	1	I	Сигнал сброса моста АНВ-АНВ
STATUS	1	O	Сигнал состояния МСК-01
COMIRQ[3:0]	4	O	Сигналы прерываний для внешнего процессора
IRQ_ALL	1	O	Прерывания, поступающие на внешний процессор, объединенные по ИЛИ с учетом маски.
Всего 17 выводов			

Таблица 7.6. UART

Наименование сигнала	Количество	Тип	Назначение
SIN	1	I	Вход последовательных данных
SOUT	1	O	Выход последовательных данных
Всего 2 вывода			

Таблица 7.7. Электропитание

Название вывода	Количество	Назначение
CVDD		Напряжение электропитания ядра и PLL
PVDD		Напряжение электропитания цифровых входных и выходных драйверов и LVDS.

GND		Земля ядра, входных и выходных драйверов, PLL и LVDS.
Всего 64 вывода		

Нумерация выводов микросхемы МСК-01 в корпусе HSBGA- 416 приведена в Таблица 7.8.

Таблица 7.8 Нумерация выводов микросхемы МСВ-01 в корпусе HSBGA-416

№ вывода корпуса	Тип вывода	Условное обозначение	№ вывода корпуса	Тип вывода	Условное обозначение
F2	I	SINP[12]	AC1	I	DINN[6]
D2	I	DINP[12]	R3	O	SOUTP[5]
C2	I	DINN[12]	T3	O	SOUTN[5]
G1	O	SOUTP[11]	V3	O	DOUTN[5]
H1	O	SOUTN[11]	U3	O	DOUTP[5]
K1	O	DOUTN[11]	Y4	I	SINN[5]
J1	O	DOUTP[11]	W4	I	SINP[5]
C1	I	SINN[11]	AA4	I	DINP[5]
B1	I	SINP[11]	AB4	I	DINN[5]
D1	I	DINP[11]	W2	O	SOUTP[4]
E1	I	DINN[11]	Y2	O	SOUTN[4]
L1	O	SOUTP[10]	AB2	O	DOUTN[4]
M1	O	SOUTN[10]	AA2	O	DOUTP[4]
P1	O	DOUTN[10]	AA3	I	SINN[4]
N1	O	DOUTP[10]	Y3	I	SINP[4]
H2	I	SINN[10]	AB3	I	DINP[4]
G2	I	SINP[10]	AC3	I	DINN[4]
J2	I	DINP[10]	AE1	I	DINN[3]
K2	I	DINN[10]	AD1	I	DINP[3]
F3	O	SOUTP[9]	AC2	I	SINP[3]
G3	O	SOUTN[9]	AD2	I	SINN[3]
J3	O	DOUTN[9]	AE5	O	DOUTP[3]
H3	O	DOUTP[9]	AF5	O	DOUTN[3]
F4	I	SINN[9]	AF4	O	SOUTN[3]
E4	I	SINP[9]	AE4	O	SOUTP[3]
G4	I	DINP[9]	AF7	I	DINN[2]
H4	I	DINN[9]	AE7	I	DINP[2]
L2	O	SOUTP[8]	AC7	I	SINP[2]
M2	O	SOUTN[8]	AD7	I	SINN[2]
P2	O	DOUTN[8]	AE6	O	DOUTP[2]
N2	O	DOUTP[8]	AF6	O	DOUTN[2]
T2	I	SINN[8]	AD6	O	SOUTN[2]
R2	I	SINP[8]	AC6	O	SOUTP[2]
U2	I	DINP[8]	AF9	I	DINN[1]
V2	I	DINN[8]	AE9	I	DINP[1]
N4	I	PLL_EN	AC9	I	SINP[1]
P4	I	XTI10	AD9	I	SINN[1]
R4	O	XTO10	AE8	O	DOUTP[1]
T4	I	XTI	AF8	O	DOUTN[1]
U4	O	SCLK	AD8	O	SOUTN[1]
V4	I	nRST	AC8	O	SOUTP[1]
L3	O	SOUTP[7]	AF11	I	DINN[0]
M3	O	SOUTN[7]	AE11	I	DINP[0]
P3	O	DOUTN[7]	AC11	I	SINP[0]
N3	O	DOUTP[7]	AD11	I	SINN[0]
K4	I	SINN[7]	AE10	O	DOUTP[0]
J4	I	SINP[7]	AF10	O	DOUTN[0]
L4	I	DINP[7]	AD10	O	SOUTN[0]
M4	I	DINN[7]	AC10	O	SOUTP[0]
T1	O	SOUTP[6]	AF15	I	nWES
U1	O	SOUTN[6]	AE15	I	nRDS
W1	O	DOUTN[6]	AD15	O	nACKS
V1	O	DOUTP[6]	AC15	I	nCSS
AA1	I	SINN[6]	AF16	I	TCK
Y1	I	SINP[6]	AE16	I	TRST
AB1	I	DINP[6]	AD16	I	TMS
AD19	-	-	AC16	I	TDI

Продолжение Таблица 7.8

№ вывода корпуса	Тип вывода	Условное обозначение	№ вывода корпуса	Тип вывода	Условное обозначение
AF17	O	TDO	P23	IO	D[7]
AE17	I	nIRQ[0]	P24	IO	D[6]
AD17	I	nIRQ[1]	R25	IO	D[5]
AC17	I	nIRQ[2]	R26	IO	D[4]
AF18	I	nIRQ[3]	N23	IO	D[3]
AE18	O	COMIRQ[0]	N24	IO	D[2]
AD18	O	COMIRQ[1]	P25	IO	D[1]
AC18	O	COMIRQ[2]	P26	IO	D[0]
AF19	O	COMIRQ[3]	N25	O	A[27]
AE19	I	nRSTM	M25	O	A[26]
AC19	O	STATUS	M24	O	A[25]
AF20	I	SIN	M23	O	A[24]
AE20	O	SOUT	L26	O	A[23]
AD20	O	SCAS	L25	O	A[22]
AC20	O	SWE	L24	O	A[21]
AF21	I	BYTE	L23	O	A[20]
AE21	O	IRQ_ALL	K26	O	A[19]
AD21	O	nCS[0]	K25	O	A[18]
AC21	O	nCS[1]	K24	O	A[17]
AF22	O	nCS[2]	K23	O	A[16]
AE22	O	nCS[3]	J26	O	A[15]
AD22	O	DQM[0]	J25	O	A[14]
AC22	O	DQM[1]	J24	O	A[13]
AE23	O	DQM[2]	J23	O	A[12]
AF23	O	DQM[3]	H26	O	A[11]
AF24	O	A10	H25	O	A[10]
AB25	O	BA[0]	H24	O	A[9]
AC25	O	BA[1]	H23	O	A[8]
AB24	O	ACKS	G26	O	A[7]
AC26	O	nWE	G25	O	A[6]
AD26	O	nRD	G24	O	A[5]
AA24	I	nACK	G23	O	A[4]
Y23	O	SRAS	F26	O	A[3]
AB26	IO	D[31]	F25	O	A[2]
Y24	IO	D[30]	F24	O	A[1]
AA25	IO	D[29]	F23	O	A[0]
AA26	IO	D[28]	E26	IO	DS[31]
W23	IO	D[27]	D26	IO	DS[30]
W24	IO	D[26]	C26	IO	DS[29]
Y25	IO	D[25]	B26	IO	DS[28]
Y26	IO	D[24]	E25	IO	DS[27]
V23	IO	D[23]	D25	IO	DS[26]
V24	IO	D[22]	C25	IO	DS[25]
W25	IO	D[21]	E24	IO	DS[24]
W26	IO	D[20]	D24	IO	DS[23]
U23	IO	D[19]	A24	IO	DS[22]
U24	IO	D[18]	E23	IO	DS[21]
V25	IO	D[17]	B23	IO	DS[20]
V26	IO	D[16]	A23	IO	DS[19]
T23	IO	D[15]	C22	IO	DS[18]
T24	IO	D[14]	B22	IO	DS[17]
U25	IO	D[13]	A22	IO	DS[16]
U26	IO	D[12]	D21	IO	DS[15]
R23	IO	D[11]	C21	IO	DS[14]
R24	IO	D[10]	B21	IO	DS[13]
T25	IO	D[9]	A21	IO	DS[12]
T26	IO	D[8]	D20	IO	DS[11]

Продолжение Таблица 7.8

№ вывода корпуса	Тип вывода	Условное обозначение	№ вывода корпуса	Тип вывода	Условное обозначение
C20	IO	DS[10]	C6	O	SOUTP[12]
B20	IO	DS[9]	B5	O	SOUTN[12]
A20	IO	DS[8]	E3	O	DOUTN[12]
D19	IO	DS[7]	D3	O	DOUTP[12]
C19	IO	DS[6]	E2	I	SINN[12]
B19	IO	DS[5]	A1		GND
A19	IO	DS[4]	A12		GND
D18	IO	DS[3]	A2		GND
C18	IO	DS[2]	AA23		GND
B18	IO	DS[1]	AB23		GND
A18	IO	DS[0]	AC14		GND
D17	I	AS[15]	AC23		GND
C17	I	AS[14]	AC24		GND
B17	I	AS[13]	AD14		GND
A17	I	AS[12]	AD23		GND
D16	I	AS[11]	AD24		GND
C16	I	AS[10]	AD25		GND
B16	I	AS[9]	AE14		GND
A16	I	AS[8]	AE24		GND
D15	I	AS[7]	AE25		GND
C15	I	AS[6]	AE26		GND
B15	I	AS[5]	AF14		GND
A15	I	AS[4]	AF26		GND
D14	I	AS[3]	AF3		GND
C14	I	AS[2]	B12		GND
B14	I	AS[1]	B2		GND
A14	I	AS[0]	B3		GND
A10	O	SOUTP[15]	C12		GND
B10	O	SOUTN[15]	C3		GND
D10	O	DOUTN[15]	C4		GND
C10	O	DOUTP[15]	D12		GND
B11	I	SINN[15]	D4		GND
A11	I	SINP[15]	D5		GND
C11	I	DINP[15]	F1		GND
D11	I	DINN[15]	K10		GND
A8	O	SOUTP[14]	K11		GND
B8	O	SOUTN[14]	K16		GND
D8	O	DOUTN[14]	K17		GND
C8	O	DOUTP[14]	K3		GND
B9	I	SINN[14]	L10		GND
A9	I	SINP[14]	L11		GND
C9	I	DINP[14]	L12		GND
D9	I	DINN[14]	L13		GND
A5	O	SOUTP[13]	L14		GND
A4	O	SOUTN[13]	L15		GND
B6	O	DOUTN[13]	L16		GND
A6	O	DOUTP[13]	L17		GND
B7	I	SINN[13]	M11		GND
A7	I	SINP[13]	M12		GND
C7	I	DINP[13]	M13		GND
D7	I	DINN[13]	M14		GND

Продолжение Таблица 7.8

№ вывода корпуса	Тип вывода	Условное обозначение	№ вывода корпуса	Тип вывода	Условное обозначение
M15		GND	A3		PVDD
M16		GND	AC12		PVDD
N11		GND	AC13		PVDD
N12		GND	AC4		PVDD
N13		GND	AC5		PVDD
N14		GND	AD12		PVDD
N15		GND	AD13		PVDD
N16		GND	AD3		PVDD
N26		GND	AD4		PVDD
P11		GND	AD5		PVDD
P12		GND	AE12		PVDD
P13		GND	AE13		PVDD
P14		GND	AE2		PVDD
P15		GND	AE3		PVDD
P16		GND	AF1		PVDD
R1		GND	AF12		PVDD
R11		GND	AF13		PVDD
R12		GND	AF2		PVDD
R13		GND	AF25		PVDD
R14		GND	B4		PVDD
R15		GND	C5		PVDD
R16		GND	D6		PVDD
T10		GND	K12		PVDD
T11		GND	K13		PVDD
T12		GND	M10		PVDD
T13		GND	M26		PVDD
T14		GND	N10		PVDD
T15		GND	P17		PVDD
T16		GND	R17		PVDD
T17		GND	U14		PVDD
U10		GND	U15		PVDD
U11		GND	A13		CVDD
U16		GND	A25		CVDD
U17		GND	A26		CVDD
W3		GND	B13		CVDD
			B24		CVDD
			B25		CVDD
			C13		CVDD
			C23		CVDD
			C24		CVDD
			D13		CVDD
			D22		CVDD
			D23		CVDD
			K14		CVDD
			K15		CVDD
			M17		CVDD
			N17		CVDD
			P10		CVDD
			R10		CVDD
			U12		CVDD
			U13		CVDD