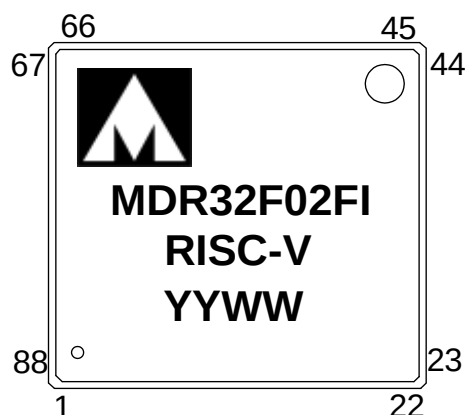




Микросхема микроконтроллера для трехфазного электросчетчика MLDR187



YY – год выпуска
WW – неделя выпуска

Основные параметры микросхемы

- Напряжение источника питания 3,0 – 3,6 В;
- 32 разрядная RISC-V архитектура BM-310S с системой команд RV32IMC;
- Встроенная память FLASH 256 + 8 Кбайт;
- Встроенная память RAM 112 Кбайт;
- Встроенная память OTP 16 Кбайт.

Обозначение	Диапазон
MDR32F02FI	минус 50 – 85 °C

Тип корпуса:

- 88-и выводной пластиковый корпус 84 LD QFN EXPOSED PAD.

Основные характеристики:

Ядро:

- 32-битное RISC-V ядро BM-310S с системой команд RV32IMC и тактовой частотой до 60 МГц;
- умножение за два цикла.

Память:

- загрузочное OTP размером 16 Кбайт;
- встроенная энергонезависимая память программ FLASH типа размером 256 Кбайт (основная область) + 8 Кбайт (информационная область);
- встроенное ОЗУ размером 112 Кбайт.

Питание и тактовая частота:

- внешнее питания 3,0...3,6 В;
- встроенный регулятор напряжения для питания ядра;
- встроенные схемы контроля питания;
- встроенный домен с батарейным питанием;
- встроенный подстраиваемый RC генератор 8 МГц;
- встроенный подстраиваемый RC генератор 32 кГц;
- внешний осциллятор 8...16 МГц;
- внешний осциллятор 32 кГц;
- встроенный умножитель тактовой частоты PLL для ядра.

Режим пониженного энергопотребления:

- батарейный домен с часами реального времени, календарём, тремя детекторами фиксации проникновения и ОЗУ 512 байт.

Аналоговые модули:

- 24-разрядный $\Sigma\Delta$ АЦП (7 независимых каналов с ПКУ);
- 10-разрядный АЦП (3 внешних мультиплицируемых канала и канал внутреннего термодатчика).

Периферия:

- контроллер прямого доступа в память с функциями передачи Периферия-Память, Память-Память;
- контроллеры интерфейсов 4 - UART, 3 - SSP, 1 - I2C, 1 - ISO7816;
- сопроцессоры блочных шифров «Кузнечик», «Магма» и AES;
- генератор случайных чисел и сопроцессоры, для обеспечения защищенного обмена данными по протоколу СПОДЭС;
- сетка, датчик частоты и напряжения питания;
- до 55 пользовательских линий ввода/вывода;
- четыре блока 32-разрядных таймеров с 4-мя каналами захвата событий и ШИМ;
- два сторожевых таймера;
- блок подсчета CRC с изменяемым полиномом.

Режим отладки:

- последовательный отладочный интерфейс JTAG.

Содержание

Содержание	3
Введение	4
Структурная схема	5
Описание выводов	6
Расположение выводов в корпусе	9
Система питания	10
Структурная схема подачи питания	11
Схема сброса при включении и выключении основного питания	12
Организация памяти	13
Загрузочное ПЗУ и режимы работы микроконтроллера	17
Контроллер FLASH памяти программ	24
Система команд	33
Процессорное ядро BM-310S	36
Блок АЦП для измерения напряжений и токов в электрической сети	39
Алгоритмы вычисления окончательных результатов и их соответствия внешним сигналам	92
Аппаратный блок вычисления CRC	101
Сигналы тактовой частоты	105
Батарейный домен и часы реального времени	117
Порты ввода-вывода	132
Детектор напряжения питания	139
Таймеры общего назначения	143
Контроллер АЦП	178
Контроллер интерфейса I2C	188
Контроллер SSP	195
Контроллер UART	227
Контроллер прямого доступа в память DMA	270
Прерывания	324
Контроллер обработки внешних прерываний	327
Подсистема отладки	332
Сторожевые таймеры	338
Блок прямого и обратного L-преобразования	346
Блок замены S-box	350
Блок байтовой замены p-byte	354
Блок битовой замены p-bit	357
Модуль обработки датчиков безопасности (SENSORS)	360
Датчик частоты (CLK_MEASURE)	365
Генератор случайных чисел (RANDOM)	369
Контроллер UART (ISO7816)	373
Модуль однократно программируемой памяти (OTP)	418
Предельно-допустимые характеристики микросхемы	426
Электрические параметры микросхемы	428
Габаритный чертеж микросхемы	430
Информация для заказа	432
Лист регистрации изменений	433

Введение

Микросхема относится к серии микроконтроллеров со встроенной Flash памятью программ и построена на базе процессорного RISC-V ядра BM-310S. Микроконтроллер работает на тактовой частоте до 60 МГц и содержит 256+8 Кбайт Flash памяти и 112 Кбайт ОЗУ. Микроконтроллер включают в себя развитую периферию для построения счетчиков электроэнергии 1- и 3-х фазных сетей. Периферия включает в себя 7 каналов для 3-фазной сети (или 3 канала для 1-фазной сети) 24-битных независимых $\Sigma\Delta$ АЦП. Каждый канал АЦП имеет предусилитель, фазовую подстройку (для коррекции фазы не хуже 0,1), а также аппаратный блок для вычисления среднеквадратического значения сигнала. Каждый канал АЦП может быть включен или отключен независимо от других каналов и имеет отдельный канал прямого доступа в память. Еще один дополнительный 10 битный АЦП последовательного приближения может быть использован для мониторинга напряжения питания основного или батарейного доменов, а также для измерения температуры или захвата внешнего сигнала. В состав микроконтроллера входит пять интерфейсов UART, три SSP и один интерфейс I2C. Криптографическая часть микросхемы включает сопроцессоры блочных шифров «Кузнечик» (ГОСТ Р34.13-2015), «Магма» и AES, генератор случайных чисел и сопроцессоры, для обеспечения защищенного обмена данными по протоколу СПОДЭС, а также инженерные методы защиты криптографической части: сетка, датчик частоты и напряжения питания. Микроконтроллер содержит четыре 32-х разрядных таймера с 4 каналами схем захвата и ШИМ с функциями формирования «мертвой зоны» и аппаратной блокировки. Так же микроконтроллер содержит системный 64-разрядный таймер и два сторожевых таймера.

Встроенные RC генераторы HSI (8 МГц) и LSI (40 кГц), внешние генераторы HSE (8...16 МГц) и LSE (32 кГц) и схема умножения тактовой частоты PLL для ядра позволяют гибко настраивать скорость работы микроконтроллеров.

Архитектура системой шины за счет регулировки частоты периферийных блоков позволяет уменьшить потребление всей системы. Контроллер DMA позволяет ускорить обмен информацией между ОЗУ и периферией без участия процессорного ядра.

Встроенный регулятор для формирования питания внутренний цифровой части формирует напряжения 1,2 В и не требует дополнительных внешних элементов. Таким образом, для работы микроконтроллера достаточно одного внешнего напряжения питания в диапазоне от 3,0 до 3,6 В. Так же в микроконтроллере реализован батарейный домен, работающий от внешней батареи при отсутствии основного питания. В батарейном домене могут быть сохранены криптографические ключи, а также работают часы реального времени. Встроенные детекторы напряжения питания могут отслеживать уровень внешнего основного питания, уровень напряжения питания на батарее. Аппаратные схемы сброса по просадке питания позволяют исключить сбойную работу микросхемы при выходе уровня напряжения питания за допустимые пределы.

Структурная схема

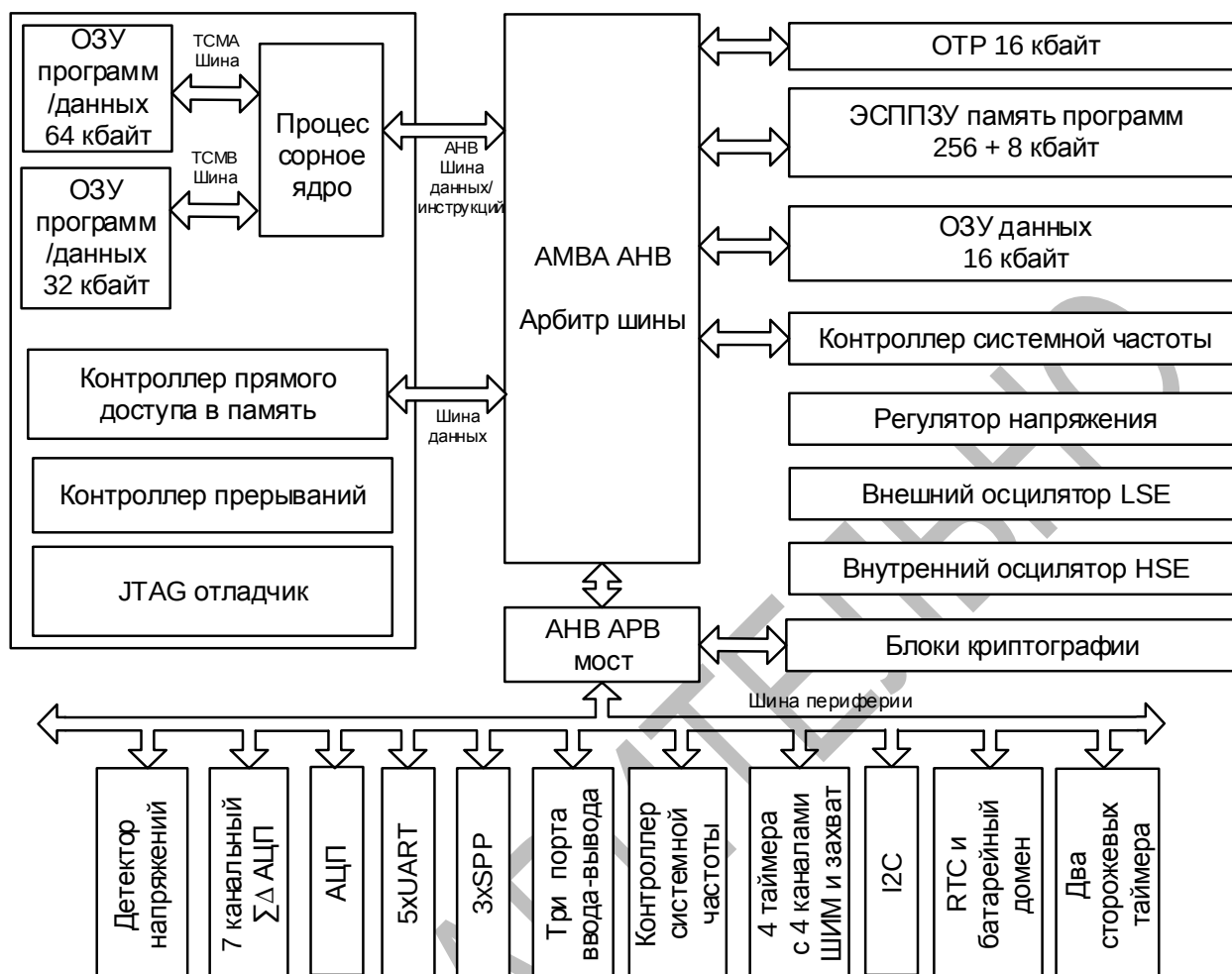


Рисунок 1. Структурная схема микросхемы

Описание выводов

Таблица 1. Описание выводов 88-и выводного корпуса

Вывод	№ вывода корпуса	Дополнительные функции вывода			
		Аналог.	Основ.	Альтер.	Переопр.
Порт А					
PA0	41	–	TMR1_CH1	SSP2_FSS	–
PA1	40	–	TMR1_CH1N	SSP2_CLK	–
PA2	39	–	TMR1_CH2	SSP2_RXD	–
PA3	38	–	TMR1_CH2N	SSP2_TXD	–
PA4	37	–	TMR1_CH3	–	–
PA5	36	–	TMR1_CH3N	–	–
PA6/TCK	29	–	TMR1_CH4	–	–
PA7/TDO	34	–	TMR1_CH4N	–	–
PA8/TMS	26	–	TMR1_ETR	–	–
PA9/TDI	32	–	TMR1_BLK	–	–
PA10	31	–	EXT_INT1	UART3_RXD	TMR4_CH1
PA11	30	–	TMR2_CH4N	UART3_TXD	TMR4_CH1N
PA12	35	–	SSP1_FSS	–	TMR4_CH2
PA13	28	–	SSP1_CLK	–	TMR4_CH2N
PA14	27	–	SSP1_RXD	–	TMR4_CH3
PA15	33	–	SSP1_TXD	–	TMR4_CH3N
Порт В					
PB0	3	–	UART1_TXD	UART7816_TXD	TMR3_CH1
PB1	4	–	UART1_RXD	UART7816_RXD	TMR3_CH1N
PB2	5	–	nSIROUT1	UART7816_CLK	TMR3_CH2
PB3	6	–	nSIRIN1	UART7816_CTS	TMR3_CH2N
PB4	7	–	nUART1DTR	–	TMR3_CH3
PB5	8	–	nUART1RTS	UART7816_RTS	TMR3_CH3N
PB6	9	–	nUART1RI	EXT_INT2	TMR3_CH4
PB7	10	–	nUART1DCD	EXT_INT3	TMR3_CH4N
PB8	12	–	nUART1DSR	TMR2_ETR	TMR3_ETR
PB9	11	–	nUART1CTS	TMR2_BLK	TMR3_BLK
PB10	13	–	TMR2_CH2	UART4_RXD	–
PB11	14	–	TMR2_CH2N	UART4_TXD	–
PB12	15	–	TMR2_CH3	–	–
PB13	16	–	TMR2_CH3N	UART2_TXD	–
PB14	17	–	TMR2_CH4	UART2_RXD	–
Порт С					
PC0/MODE0	51	–	TMR4_ETR	–	TMR4_CH4
PC1/MODE1	52	–	TMR4_BLK	–	TMR4_CH4N
PC2	53	–	TMR2_CH1	SSP3_FSS	I2C1_SCL
PC3	54	–	TMR2_CH1N	SSP3_CLK	I2C1_SDA
PC4	55	–	EXT_INT2	SSP3_RXD	–
PC5	56	–	EXT_INT3	SSP3_TXD	–
PC6	57	–	TMR2_ETR	I2C1_SCL	–
PC7	58	–	TMR2_BLK	I2C1_SDA	–
Порт D					
PD0	18	–	SSP2_FSS	–	–
PD1	19	–	SSP2_CLK	–	–
PD2	20	–	SSP2_RXD	–	–

Вывод	№ вывода корпуса	Дополнительные функции вывода			
		Аналог.	Основ.	Альтер.	Переопр.
PD3	21	—	SSP2_TXD	—	—
PD4	22	—	UART2_TXD	—	—
PD5	44	—	UART2_RXD	—	—
PD6	43	—	UART3_RXD	—	—
PD7	42	—	UART3_TXD	—	—
PD8	25	—	UART4_RXD	—	—
PD9	24	—	UART4_TXD	—	—
PD10	23	—	UART7816_TXD	—	—
PD11	45	—	UART7816_RXD	—	—
PD12	46	—	UART7816_CLK	—	—
PD13	47	—	UART7816_CTS	—	—
PD14	48	—	UART7816_RTS	—	—
PD15	49	—	EXT_INT1	LSE_OUT	—
Порт Σ АЦП					
VR_2p4V	83				
I0P	69				
I0N	70				
V0P	71				
V0N	72				
I3P	73				
I3N	74				
I1P	75				
I1N	76				
V1P	77				
V1N	78				
I2P	79				
I2N	80				
V2P	81				
V2N	82				
Порт SAR АЦП					
SAR_AIN1	84				
SAR_AIN2	85				
SAR_AIN3	86				
Системное управление					
nRESET	59	Сигнал внешнего сброса			
WAKEUP1	63	Сигнал вскрытия и внешнего			
WAKEUP2	62	Сигнал вскрытия и внешнего			
WAKEUP3	61	Сигнал вскрытия и внешнего			
OSC_IN	87	Вход генератора HSE			
OSC_OUT	88	Выход генератора HSE			
OSC_IN32	67	Вход генератора LSE			
OSC_OUT32	66	Выход генератора LSE			
SW	64	Выход управления внешним ключом питания BUCC на UCC			
Питание					
UCC	1, 50	Питание 3,0...3,6			
BUCC	65	Батарейное питание 1.8...3,6 В			
GND	2	Общий			
EXPOSED PAD		Общий			

Вывод	№ вывода корпуса	Дополнительные функции вывода			
		Аналог.	Основ.	Альтер.	Переопр.
AUCC	68	Аналоговое питание Σ ЛАЦП и SAR АЦП 2,4...3,6 В			
Выводы для тестирования и исследования					
JTAG_EN	60	Вход разрешения отладочного TAP интерфейса на выводах PB[9..6]. Следует подсоединять к выводу Общий.			

ПРЕДВАРИТЕЛЬНО

Расположение выводов в корпусе

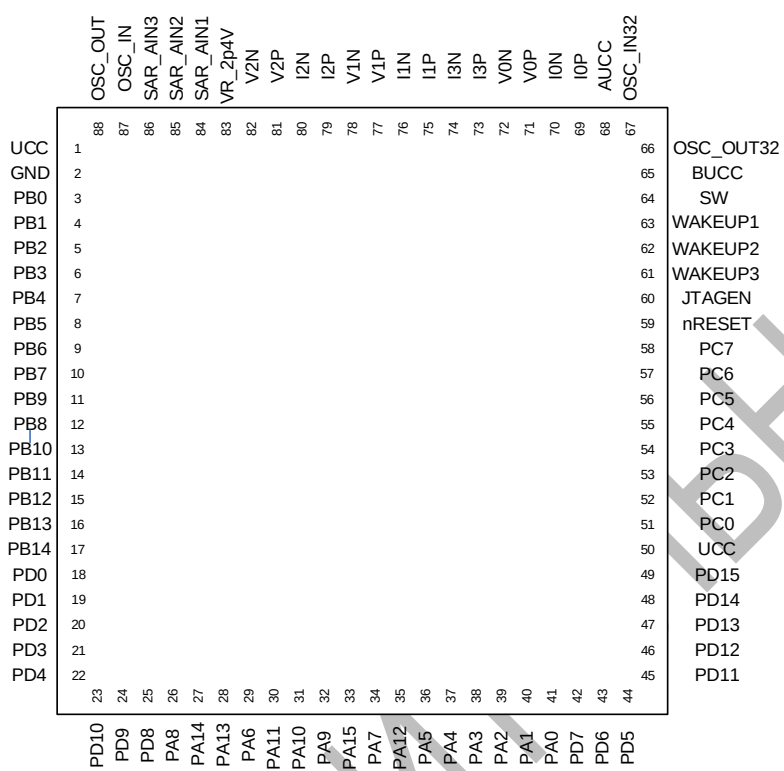


Рисунок 2. Микросхема в 88-выводном пластиковом корпусе 88 LD QFN EXPOSED PAD

Система питания

Микроконтроллер этой серии имеет несколько типов выводов питания.

Ucc выводы: Основное питание микросхемы, включает питание пользовательских выводов, встроенного регулятора напряжения, PHU, PLL, генераторов, компаратора и АЦП последовательного приближения.

BUcc вывод: Питание батарейного домена используется при отсутствии основного питания Ucc для питания батарейного домена и LSE генератора. Переключение с основного питания на батарейное происходит автоматически при снижении уровня Ucc ниже 2,0 В. Переключение с батарейного питания на основное происходит автоматически спустя примерно 4 мс после превышения уровнем Ucc значения 2,0 В. Если в системе не требуется батарейного питания, вывод BUcc должен быть объединен с Ucc.

AUcc выводы: Питание аналоговых блоков сигма дельта АЦП и формирователя опоры выведено на отдельные выводы для уменьшения помех создаваемых работой других блоков. На данные выводы должно подаваться напряжения с того же источника что и Ucc, но при этом на печатной плате должны быть применены меры по снижению наводки помех.

GND выводы: Основная земля питания.

EXPOSED PAD: Земля аналогового питания AUcc и цифрового питания Ucc. Данные выводы должны соединяться с GND.

Структурная схема подачи питания

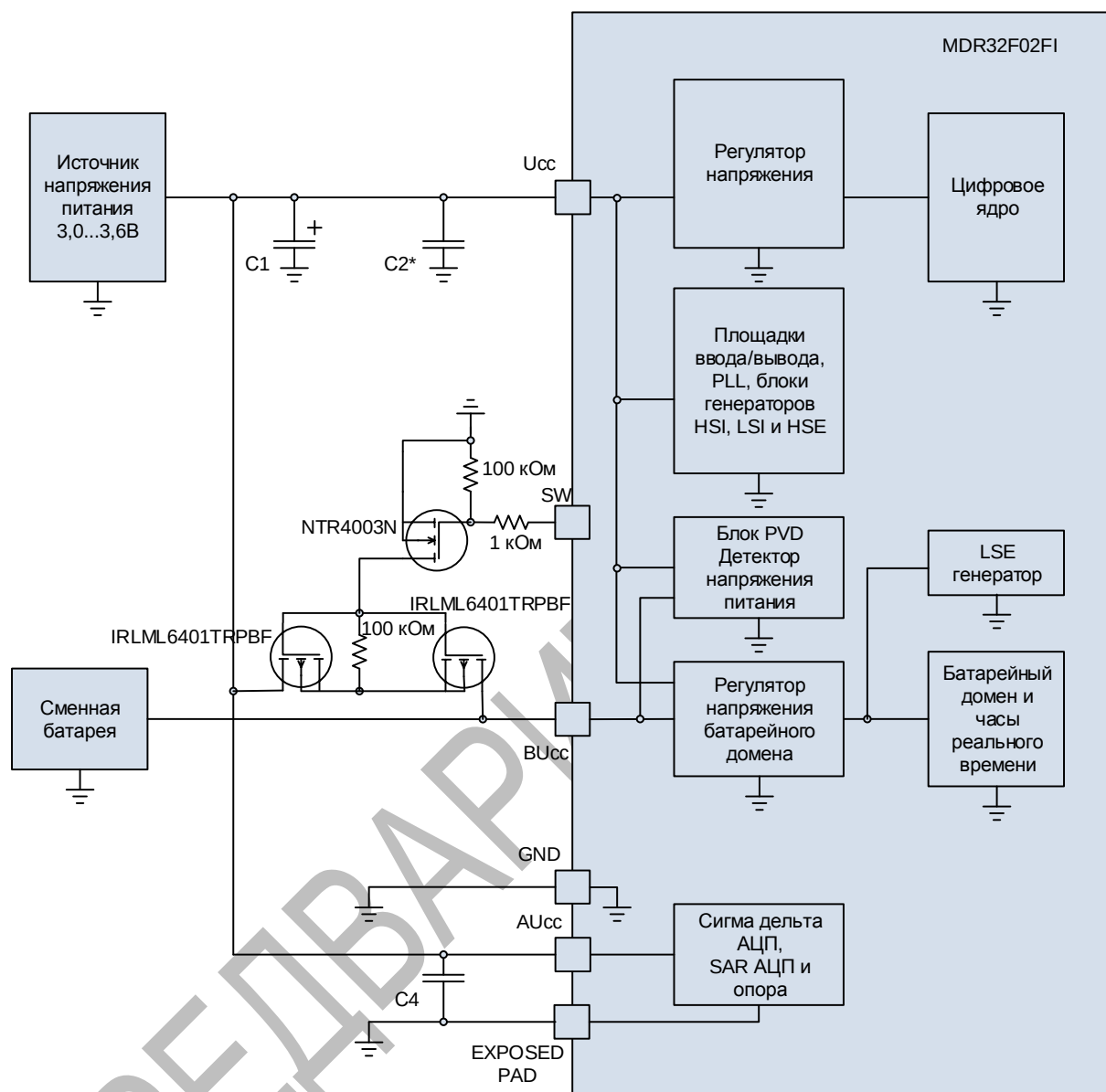


Рисунок 3. Структурная схема подачи питания

Примечание:

1. Конденсатор $C1 = 22 \text{ мкФ}$, $C2 = C3 = C4 = C5 = 0,1 \text{ мкФ}$.
* - конденсаторы должны быть установлены у каждого вывода питания.
2. При отсутствии батареи, вывод BU_{cc} должен быть объединен с U_{cc}
3. Если используется АЦП, то напряжение питания U_{cc} (AU_{cc}) должно быть в пределах от 2,4 до 3,6 В;

Микроконтроллер имеет встроенный детектор напряжения питания, подробнее см. раздел «Детектор напряжения питания».

Схема сброса при включении и выключении основного питания

При включении питания вырабатывается внутренний сигнал сброса POR для цифровой части, питание U_{cc} нарастает и, пока оно не превысило уровень 2,0 В, сигнал сброса POR удерживается; после превышения данного уровня сигнал POR выдается еще на протяжении ~ 4 мс для того, чтобы гарантированно установилось напряжение питания, после чего сигнал POR снимается, и схема может начать работать.

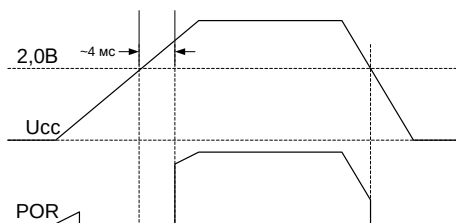


Рисунок 4. Диаграмма формирования сброса микроконтроллера

При снижении напряжения питания U_{cc} ниже уровня 2,0 В сигнал POR вырабатывается без задержки.

Сигнал POR так же служит для переключения питания батарейного домена между Висс и U_{cc} .

При включении основного напряжения питания U_{cc} автоматически включается встроенный регулятор напряжения для формирования напряжения питания цифрового ядра. В ходе работы микроконтроллера встроенный регулятор может быть отключен.

Начальная установка микроконтроллера может быть произведена внешним сигналом сброса nRESET, внутренними сигналами сброса сторожевых таймеров или программным сбросом. При этом сигнал nRESET формируется специальной схемой сброса, содержащей фильтр «иголок» и одновибратор для увеличения длительности этого сигнала.

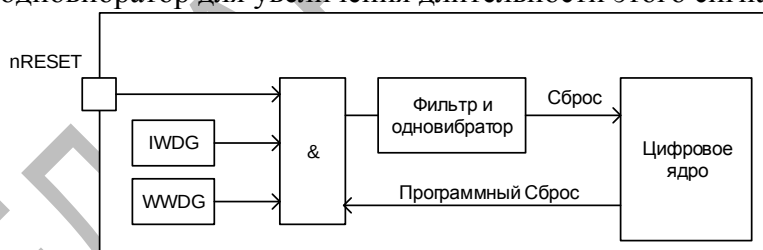


Рисунок 5. Блок-схема формирования сброса микроконтроллера

При подаче на вход nRESET импульсов сброса длительностью менее 10 нс эти импульсы отфильтровываются и не приводят к сбросу процессора. Если длительность импульса больше 200 нс, вырабатывается сигнал сброса. При этом длительность сформированного сигнала сброса будет не менее 20 мкс.

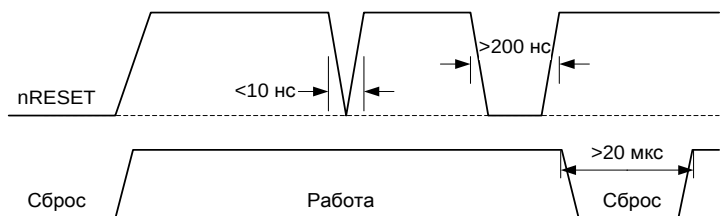


Рисунок 6. Диаграмма фильтрации помех при формировании сброса

Организация памяти

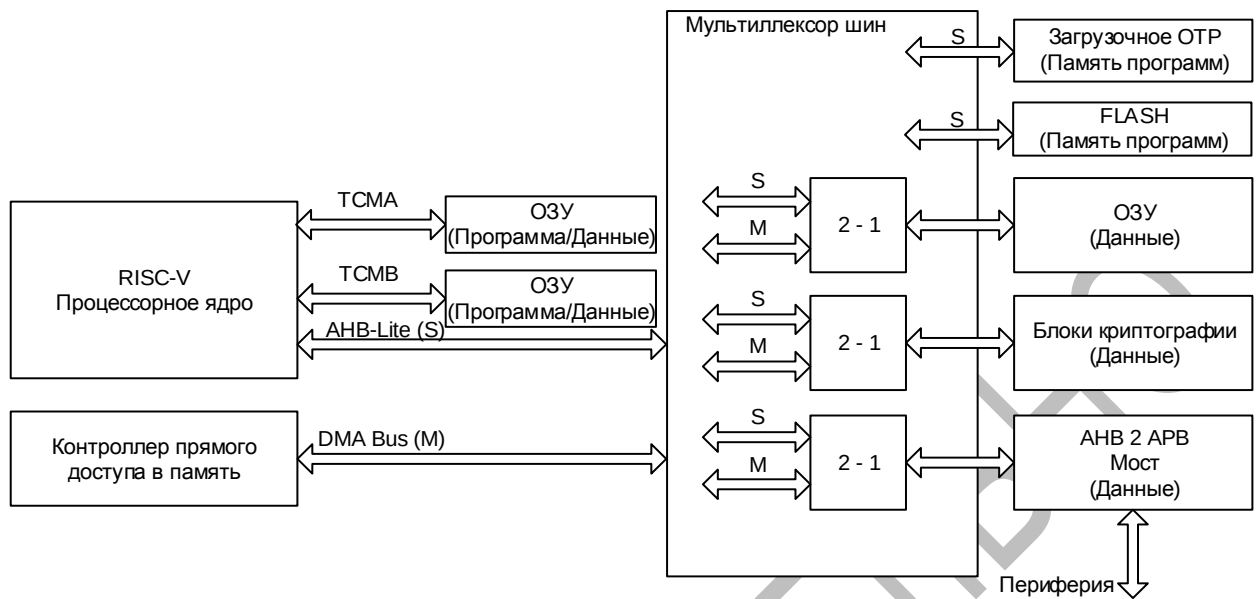


Рисунок 7. Структурная схема организации памяти

Процессорное ядро имеет две системные шины:

АМВ-Lite – шина выборки инструкций и данных.

ТСМА, ТСМВ – шина выборки инструкций/данных.

Также в микроконтроллере реализован контроллер прямого доступа в память (DMA), осуществляющий выборку через шину DMA Bus.

Все адресное пространство микроконтроллера разделено на пять основных секций: Debug, CLINT, PLIC, I/O и TCM. В данное адресное пространство отображаются различные модули памяти и периферии.

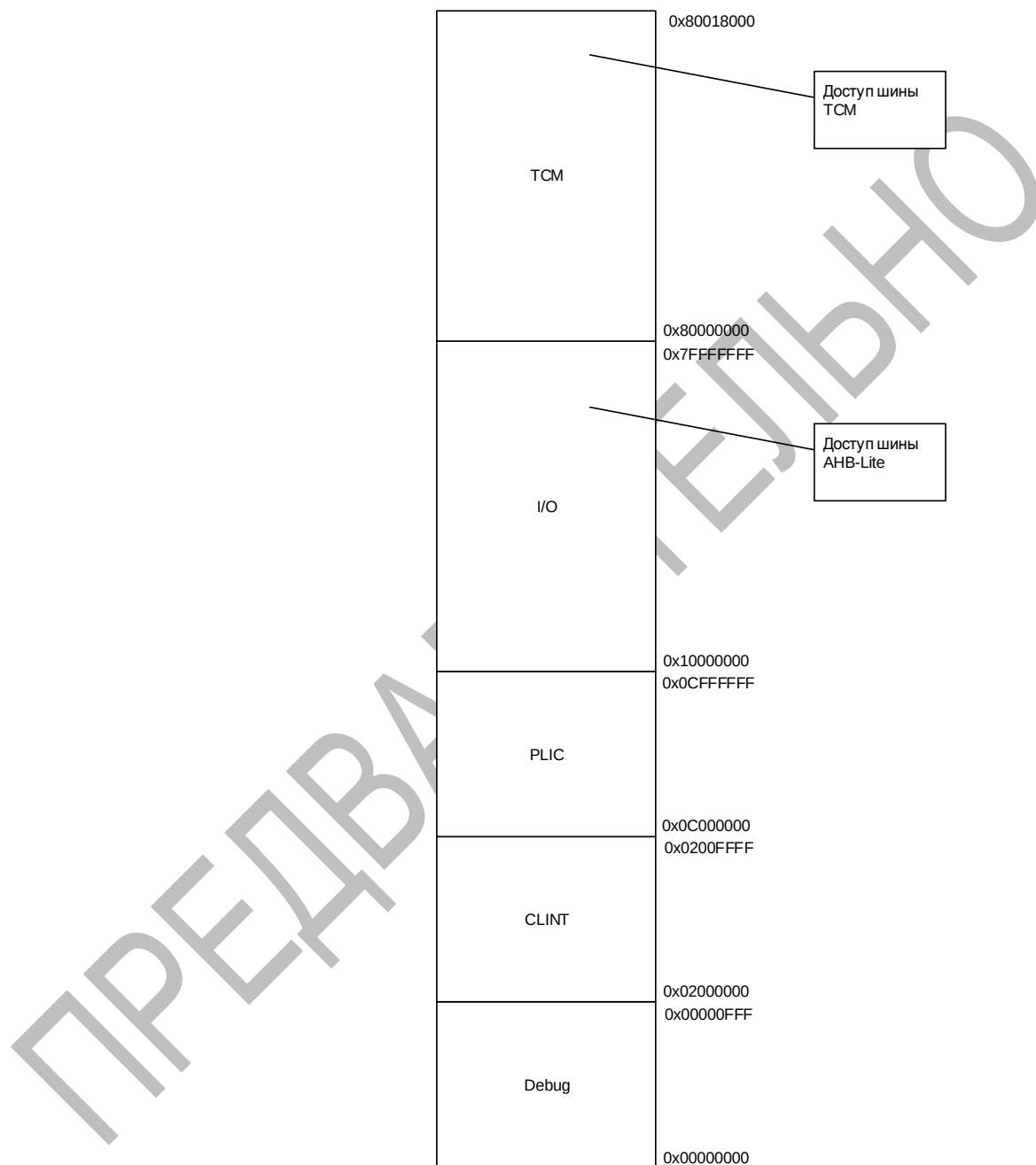


Рисунок 8. Структура адресного пространства микроконтроллера

Базовые адреса процессора

Таблица 2. Базовые адреса процессора

Адрес	Размер	Блок	Примечание
Память программ			
0x0002_0000		BOOT OTP	Загрузочная программа
0x1000_0000		FLASH	Область Flash памяти программ с пользовательской программой
Память данных			
0x2000_0000		SYSTEM RAM	Область внутреннего ОЗУ шины АНВ
Периферия			
0x4000_0000		SSP1	Регистры контроллера интерфейса SSP1
0x4000_8000		UART1	Регистры контроллера интерфейса UART1
0x4001_0000		UART2	Регистры контроллера интерфейса UART2
0x4001_8000		EEPROM_CNTRL	Регистры контроллера Flash памяти программ
0x4002_0000		RST_CLK	Регистры контроллера сигналов тактовой частоты
0x4002_8000		DMA	Регистры контроллера прямого доступа в память
0x4003_0000		I2C	Регистры контроллера интерфейса I2C
0x4003_8000		UART3	Регистры контроллера интерфейса UART3
0x4004_0000		ADC	Регистры управления АЦП
0x4004_8000		WWDG	Регистры контроллера сторожевого таймера WWDG
0x4005_0000		IWDG	Регистры контроллера сторожевого таймера IWDG
0x4005_8000		POWER	Регистры детектора напряжения питания
0x4006_0000		BKP	Регистры доступа и управления батарейным доменом
0x4006_8000		ADCIU	Регистры управления $\Sigma\Delta$ АЦП
0x4007_0000		TIMER1	Регистры управления Таймер 1
0x4007_8000		TIMER2	Регистры управления Таймер 2
0x4008_0000		PORTA	Регистры управления порта А
0x4008_8000		PORTB	Регистры управления порта В
0x4009_0000		PORTC	Регистры управления порта С
0x4009_8000		CRC	Регистры управления аппаратного блока вычисления CRC
0x400A_0000		-	-
0x400A_8000		SENSORS	Регистры блока обработки датчиков безопасности
0x400B_0000		CLK_MEASURE	Регистры блока датчика частоты
0x400B_8000		RANDOM	Регистры блока генератора случайных чисел
0x400C_0000		ISO7816	Регистры контроллера интерфейса UART (ISO7816)
0x400C_8000		SSP2	Регистры контроллера интерфейса SSP2
0x400D_0000		SSP3	Регистры контроллера интерфейса SSP3
0x400D_8000		TIMER3	Регистры управления Таймер 3

0x400E_0000		TIMER4	Регистры управления Таймер 4
0x400E_8000		UART4	Регистры контроллера интерфейса UART4
0x400F_0000		PORTD	Регистры управления порта D
CRYPTO REGION			
0x6000_0000		-	-
0x6000_4000		L_BLOCK	Регистры блока прямого и обратного L преобразования
0x6000_8000		S_BLOCK0	Регистры блока замены S-box0
0x6000_C000		S_BLOCK1	Регистры блока замены S-box1
0x6001_0000		S_BLOCK2	Регистры блока замены S-box2
0x6001_4000		S_BLOCK3	Регистры блока замены S-box3
0x6001_8000		S_BLOCK4	Регистры блока замены S-box4
0x6001_C000		S_BLOCK5	Регистры блока замены S-box5
0x6002_0000		S_BLOCK6	Регистры блока замены S-box6
0x6002_4000		S_BLOCK7	Регистры блока замены S-box7
0x6002_8000		P_BYTE	Регистры блока байтовой замены p-byte
0x6002_C000		P_BIT0	Регистры блока битовой замены p-bit0
0x6003_0000		P_BIT1	Регистры блока битовой замены p-bit1
0x6003_4000		P_BIT2	Регистры блока битовой замены p-bit2
0x6003_8000		P_BIT3	Регистры блока битовой замены p-bit3
0x7000_0000		OTP	Регистры управления блоком OTP и адресное пространство однократно программируемой памяти
SYSTEM REGION			
0x8000_0000		SYSTEM RAM	Область внутреннего ОЗУ шины TCMA
0x8001_0000		SYSTEM RAM	Область внутреннего ОЗУ шины TCMB

Загрузочное ПЗУ и режимы работы микроконтроллера

Загрузочная программа хранится в OTP. Если OTP не было запрограммировано при поставке микросхемы, то можно самостоятельно разработать код загрузочной программы и загрузить его в OTP. Ниже приводится описание одного из возможных вариантов загрузочной программы. После включения питания и снятия внутренних (POR) и внешних (nRESET) сигналов сброса, микроконтроллер начинает выполнять программу из загрузочной области BOOT ROM. В загрузочной программе микроконтроллер определяет, в каком из режимов он будет функционировать и переходит в этот режим. Режим функционирования определяется внешними выводами MODE[1:0] (PC[0], PC[1]). Так же устанавливается бит prog в регистре BKP_LDO, который может быть сброшен только при отключении основного питания Uсс. После перезапуска микроконтроллера уровни на выводах MODE[1:0] не влияют на режим функционирования микроконтроллера, если установлен бит prog, так как в этом случае режим работы будет прочитан из бит mode[1:0] регистра BKP_LDO.

В пользовательской программе выводам PC[0], PC[1] пользователем могут присваиваться функции самостоятельно.

Таблица 3. Режимы работы микроконтроллера

MODE[1:0]	Режим	Стартовый адрес/ таблица векторов прерываний	Описание
00	Микроконтроллер в режиме отладки	0x10000000	Процессор начинает выполнять программу из внутренней OTP, а затем передаёт управление во FLASH память программ. При этом разрешается работа отладочного интерфейса JTAG в случае отсутствия блокировки в OTP памяти.
01 или 10	UART загрузчик	Определяется пользователем	Микроконтроллер через интерфейс UART1 на выводах PB[1], PB[0] получает код программы в ОЗУ для исполнения
11	Микроконтроллер в режиме отладки	0x80000000	Процессор начинает выполнять программу из внутренней OTP, а затем передаёт управление в ОЗУ память программ. При этом разрешается работа отладочного интерфейса JTAG в случае отсутствия блокировки в OTP памяти.

При работе в режиме отладки разрешается работа отладочного интерфейса JTAG. При этом к микроконтроллеру может быть подключен JTAG адаптер с помощью которого программные средства разработки позволяют работать с микроконтроллером в отладочном режиме.

В отладочном режиме можно:

- стирать, записывать, считывать внутреннюю FLASH память программ;
- считывать и записывать содержимое ОЗУ, периферии;
- выполнять программу в пошаговом режиме;
- запускать программу в нормальном режиме;
- останавливать программу по точкам остановки;

- просматривать переменные выполняемой программы;
- проводить трассировку хода выполнения программного обеспечения.

Биты OTP 131064 – 131067 (последний байт) позволяют установить защищённые адресные пространства FLASH памяти. Установка любого из этих бит запрещает доступ к OTP, периферийным блокам криптографии, защищённым адресным пространствам FLASH и блокирует отладочный режим JTAG при выполнении программы из незащищённого адресного пространства FLASH и любого ОЗУ. При выполнении программы из OTP или защищённого адресного пространства FLASH, выбранного одним или несколькими вышеописанными битами, доступ и отладочный режим не блокируются.

Бит 131064 отвечает за выбор защищённого адресного пространства FLASH памяти с диапазоном адресов 0x10000000-0x1000FFFF.

Бит 131065 отвечает за выбор защищённого адресного пространства FLASH памяти с диапазоном адресов 0x10010000-0x1001FFFF.

Бит 131066 отвечает за выбор защищённого адресного пространства FLASH памяти с диапазоном адресов 0x10020000-0x1002FFFF.

Бит 131067 отвечает за выбор защищённого адресного пространства FLASH памяти с диапазоном адресов 0x10030000-0x1003FFFF.

UART загрузчик

Режим UART загрузчика предоставляет достаточный набор операций, необходимых для записи в ОЗУ какой-либо программы (в частности программатора Flash-памяти), верификации ее и запуска на выполнение. Кроме того, существует возможность задания внешним устройством скорости обмена. Помимо доступа к ОЗУ может быть осуществлен доступ и к другим адресным диапазонам (FLASH, OTP, Периферия).

В качестве источника тактовой частоты UART1 используется внутренний RC-генератор HSI с частотой 8 МГц. Так как имеется разброс значений частоты HSI, то требуется этап подбора значения делителя частоты UART1 для синхронизации с внешним устройством.

Параметры связи по UART

Для связи по UART выбраны следующие параметры канала связи:

Начальная скорость – 9600 бод.

Количество бит данных – 8.

Четность – нет.

Количество Stop бит – 1.

Загрузчик не использует FIFO UART1.

Загрузчик всегда выступает в качестве Slave, а внешнее устройство, подающее команды – в качестве Master.

Данные передаются младшим битом вперед.

Протокол обмена по UART

После синхронизации с Master загрузчик переходит в диспетчер команд. Таким образом, Master-у доступны следующие команды:

Команды UART загрузчика

Таблица 4. Команды UART загрузчика

Команда	Код	ASCII Символ	Описание
CMD_SYNC	0x00		Пустая команда. Загрузчик ее принимает, но ничего по ней не делает
CMD_CR	0x0D		Выдача приглашения Master-у
CMD_BAUD	0x42	'B'	Установка скорости обмена
CMD_LOAD	0x4C	'L'	Загрузка массива байт
CMD_VFY	0x59	'Y'	Выдача массива байт
CMD_RUN	0x52	'R'	Запуск программы на выполнение

Синхронизация с внешним устройством

Начальные условия.

На этапе синхронизации с внешним устройством (Master) вывод Rx используется как вход. Master постоянно посылает в канал синхросимвол – 0. Загрузчик подстраивает свою скорость таким образом, чтобы минимизировать ошибки обмена. Как только Загрузчик настроил скорость, он переходит в диспетчер команд и выдает приглашение (3 байта 0x0D (перевод строки), 0x0A (возврат каретки), 0x3E ('>'),) Master-у.

Master завершает выдачу синхросимволов и теперь может подавать команды, согласно протоколу обмена.

Команда CMD_SYNC

Пустая команда.

Загрузчик (Slave) ее принимает, но ничего по ней не делает. Код команды соответствует символу синхронизации.

Таблица 5. Команда CMD_SYNC

Код команды	CMD_SYNC = 0x00
ASCII символ, соответствующий коду команды	нет
Количество параметров команды	0
Формат команды:	
Master: Выдает код команды CMD_SYNC.	Slave: если команда принята с ошибками, то выдает сообщение об ошибке 0x45 ('E') и завершает обработку текущей команды

Команда CMD_CR

Выдача приглашения Master-у.

Таблица 6. Команда CMD_CR

Код команды	CMD_CR = 0x0D
ASCII символ, соответствующий коду команды	нет
Количество параметров команды	0
Формат команды:	
Master: Выдает код команды CMD_CR.	Slave: если команда принята с ошибками, то выдает сообщение об ошибке 0x45 ('E') и завершает обработку текущей команды. Выдает код команды CMD_CR. Выдает код 0x0A. Выдает код 0x3E (ASCII символ '>').

Команда CMD_BAUD

Установка скорости обмена.

Таблица 7. Команда CMD_BAUD

Код команды	CMD_BAUD = 0x42
ASCII символ, соответствующий коду команды	'B'
Количество параметров команды	1
Параметр	Новое значение скорости обмена [бод]
Формат команды:	
Master: Выдает код команды CMD_BAUD	Slave: если команда принята с ошибками, то выдает сообщение об ошибке 0x45 ('E') и завершает обработку текущей команды.
Master: Выдает параметр	Slave: Если параметр принят с ошибками, то выдает код ошибки ERR_CHN или ERR_BAUD и завершает обработку текущей команды. Выдает код команды CMD_BAUD. Устанавливает новое значение скорости обмена

Команда CMD_LOAD

Загрузка массива байт в память микроконтроллера.

Таблица 8. Команда CMD_LOAD

Код команды	CMD_LOAD = 0x4C
ASCII символ, соответствующий коду команды	'L'
Количество параметров команды	2
Параметр 1.	Адрес памяти приемника данных.
Параметр 2.	Размер массива в байтах
Формат команды:	
Master: Выдает код команды CMD_LOAD	Slave: если команда принята с ошибками, то выдает сообщение об ошибке 0x45 ('E') и завершает обработку текущей команды.
Master: Выдает параметр 1.	Slave: если хотя бы один из параметров принят с ошибками, то выдает сообщение об ошибке 0x45 ('E') и завершает обработку текущей команды.
Master: Выдает параметр 2.	Slave: если хотя бы один из параметров принят с ошибками, то выдает сообщение об ошибке 0x45 ('E') и завершает обработку текущей команды. Выдает код команды CMD_LOAD.
Master: Выдает массив байт младшим байтом вперед.	Slave: принимает массив байт. Если хотя бы один байт принят с ошибками, то выдает сообщение об ошибке и завершает обработку текущей команды, не дожидаясь окончания принятия всего массива. По окончании принятия массива выдает код ответа REPLY_OK = 0x4B ('K').

Команда CMD_VFY

Выдача массива байт из памяти микроконтроллера.

Таблица 9. Команда CMD_VFY

Код команды	CMD_VFY = 0x59
ASCII символ, соответствующий коду команды	'Y'
Количество параметров команды	2
Параметр 1	Адрес памяти источника данных
Параметр 2	Размер массива в байтах
Формат команды:	
Master: Выдает код команды CMD_VFY	Slave: если команда принята с ошибками, выдает сообщение об ошибке 0x45 ('E') и завершает обработку текущей команды
Master: Выдает параметр 1	Slave: если хотя бы один из параметров принят с ошибками, выдает сообщение об ошибке 0x45 ('E') и завершает обработку текущей команды
Master: Выдает параметр 2	Slave: если хотя бы один из параметров принят с ошибками, то выдает сообщение об ошибке 0x45 ('E') и завершает обработку текущей команды. Выдает код команды CMD_VFY. Выдает массив байт младшим байтом вперед. По окончании передачи массива выдает код ответа REPLY_OK = 0x4B ('K')

Команда CMD_RUN

Запуск программы на выполнение.

Таблица 10. Команда CMD_RUN

Код команды	CMD_RUN = 0x52
ASCII символ, соответствующий коду команды	'R'
Количество параметров команды	1
Параметр.	Адрес таблицы векторов загруженной программы
Формат команды:	
Master: Выдает код команды CMD_RUN.	Slave: если команда принята с ошибками, то выдает сообщение об ошибке 0x45 ('E') и завершает обработку текущей команды
Master: Выдает параметр.	Slave: если параметр принят с ошибками, то выдает сообщение об ошибке и завершает обработку текущей команды. Выдает код команды CMD_RUN. Устанавливает значение MSP и PC согласно таблице векторов (NVIC не перепрограммируется) и, таким образом, Slave завершает свое выполнение

Прием параметров команды

Параметры команд – это 4-х байтные числа.

Параметры передаются младшим байтом вперед.

В качестве значения параметра запрещено использовать число 0xFFFFFFFF.

Если при приеме параметра обнаружена аппаратная ошибка (UART установил в '1' какой-либо из флагов ошибки), то прием параметров не прекращается.

Анализ всех видов ошибок, связанных с передачей параметров, загрузчик производит только после принятия всех параметров команды.

Сообщения об ошибках

Сообщения об ошибках – это символ 0x45 ('E').

После выдачи сообщения об ошибке загрузчик переходит в режим ожидания следующей команды, поэтому Master после получения такого сообщения должен прекратить передачу байт, относящихся к текущей команде.

После принятия сообщения об ошибке Master должен подавать команду CMD_CR до тех пор, пока не получит корректный ответ, соответствующий этой команде.

Ошибка ERR_CHN

Аппаратная ошибка UART.

Код ошибки 0x69 ('i').

Выдается, если UART установил в '1' один из аппаратных флагов ошибки при приеме очередного байта.

Ошибка ERR_CMD

Принята неизвестная команда.

Код ошибки 0x63 ('c').

Выдается диспетчером команд, если принят неизвестный код команды.

Ошибка ERR_BAUD

Принята неизвестная команда.

Код ошибки 0x62 ('b').

Выдается диспетчером команд, если по принятому от Master-а значению скорости обмена невозможно вычислить корректное значение делителя частоты UART.

Контроллер FLASH памяти программ

Микроконтроллер содержит встроенную Flash память программ с объемом 256 Кбайт основной памяти программ и 8 Кбайт информационной памяти. В обычном режиме (бит CON = 0, регистр EEPROM_CMD) доступна основная память программ через системную шину для выборки инструкций и данных кода программы. В режиме программирования (бит CON=1, регистр EEPROM_CMD) основная и информационная память доступны как периферийное устройство и могут быть использованы для нужд разработчика приложения. В режиме программирования программный код должен выполняться из области системной шины или ОЗУ. Выполнение программного кода из Flash памяти программ в режиме программирования невозможно.

Работа Flash памяти программ в обычном режиме

Скорость доступа во Flash память ограничена и составляет порядка 30 нс, в результате, выдача новых значений из Flash памяти может происходить с частотой не более 30 МГц. Для того, чтобы процессорное ядро могло получать новые инструкции на больших частотах на каждом такте из Flash памяти извлекаются избыточные 4 байта. И пока ядро выполняет инструкцию, из памяти извлекается следующая порция данных. Таким образом, тактовая частота может превышать частоты извлечения данных из памяти в два раза при линейном выполнении программы. При возникновении переходов в выполнении программы, когда из памяти программ не выбраны нужные инструкции возникает пауза в несколько тактов процессора для того, чтобы данные успели считаться из Flash. Число тактов паузы зависит от тактовой частоты процессора, так при работе с частотой ниже 30 МГц пауза не требуется, так как Flash память успевает выдать новые данные за один такт, при частоте от 30 до 60 МГц требуется один такт паузы. Число тактов паузы задается в регистре EEPROM_CMD битами Delay[1:0]. В таблице приведены характеристики необходимой паузы для работы Flash памяти программ.

Таблица 11. Характеристики паузы для работы Flash-программ

Delay [1:0]	Тактов паузы	Тактовая частота	Примечание
0x00	0	До 30 МГц	
0x01	1	До 60 МГц	

Число тактов паузы устанавливается до момента повышения тактовой частоты или после снижения тактовой частоты.

Работа Flash памяти программ в режиме программирования

Перед переводом памяти в режим программирования необходимо в регистр EEPROM_KEY записать комбинацию 0x8AAA5551. В режиме программирования Flash память программ не может выдавать инструкции и данные процессору, поэтому перевод памяти в режим программирования (установка бита CON = 1) возможен только программой, исполняемой из ОЗУ.

В режиме программирования возможны следующие операции как с основной (бит IFREN = 0, регистр EEPROM_CON), так и с информационной (бит IFREN = 1) памятью:

- стирание всей памяти или 256 Кбайт;
- стирание страницы памяти размером 4 Кбайта;
- запись 32-битного слова в память;
- чтение 32-битного слова из памяти.

Структура памяти представлена на рисунке.

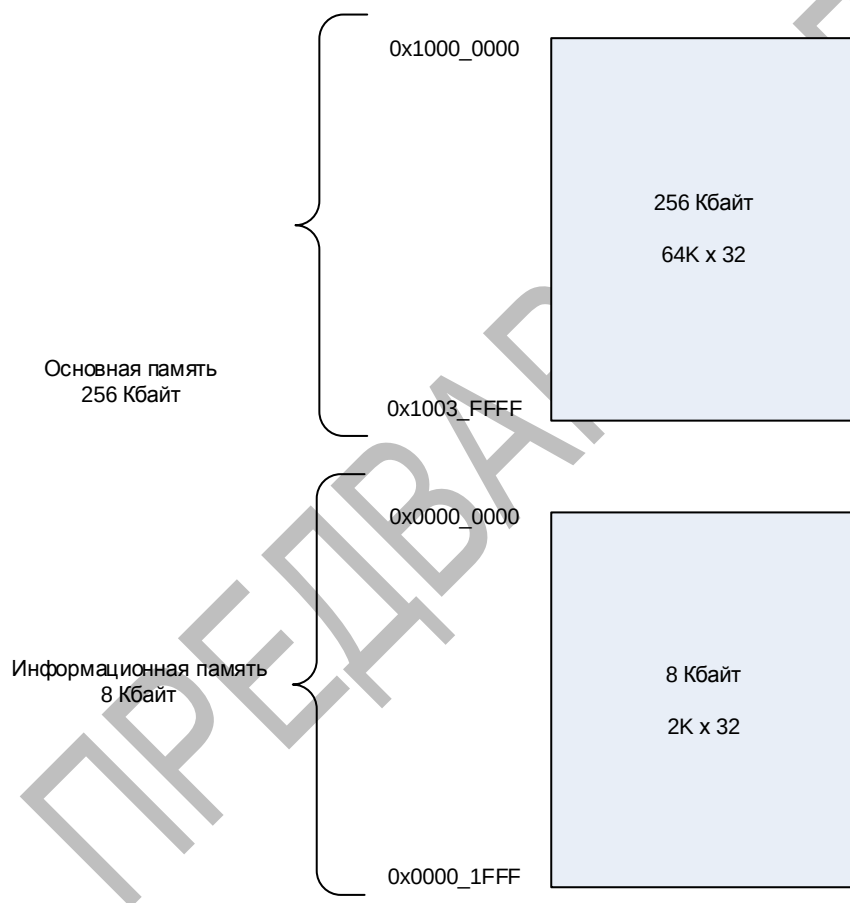


Рисунок 9. Структура памяти

Стирание всей памяти или 256 Кбайт основной памяти.

Стирание памяти возможно только в режиме программирования. Для стирания всей памяти надо установить необходимое значение в бит IFREN (1 – для всей памяти или 0 – для основной памяти), затем установить биты XE, MAS1 и ERASE в единицу, и спустя время $T_{nvs} = 5$ мкс установить бит NVSTR в единицу. Полное стирание памяти длится время $T_{me} = 40$ мс. Спустя это время необходимо очистить бит ERASE, и спустя время $T_{nvhl} = 100$ мкс очистить биты XE, MAS1 и NVSTR. Последующие операции с память можно выполнять спустя время $T_{rcv} = 10$ мкс. Временная диаграмма стирания памяти представлена на рисунке.

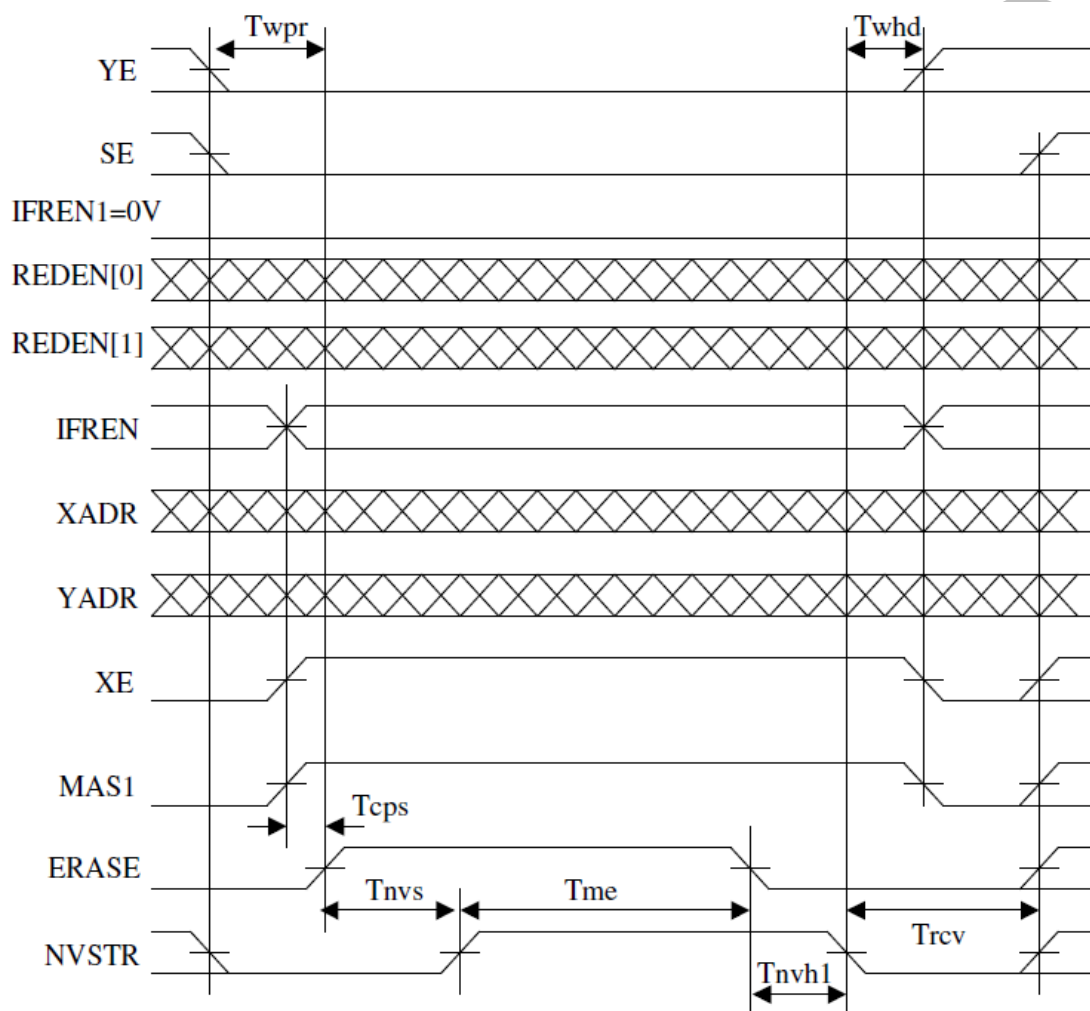


Рисунок 10. Временная диаграмма стирания памяти

Стирание страницы памяти размером 4 Кбайт.

Стирание страницы памяти возможно только в режиме программирования. Для стирания страницы памяти надо установить необходимое значение в бит IFREN (1 – для информационной памяти и 0 – для основной памяти), затем установить адрес стираемой страницы в регистре EEPROM_ADR и установить биты XE и ERASE в единицу, и спустя время $T_{nvs} = 5$ мкс установить бит NVSTR в единицу. Стирание страницы памяти длится время $T_{erase} = 40$ мс. Спустя это время необходимо очистить бит ERASE, и спустя время $T_{nvh} = 5$ мкс очистить биты XE и NVSTR. Последующие операции с памятью можно выполнять спустя время $T_{rcv} = 10$ мкс. Временная диаграмма стирания страницы памяти представлена на рисунке.

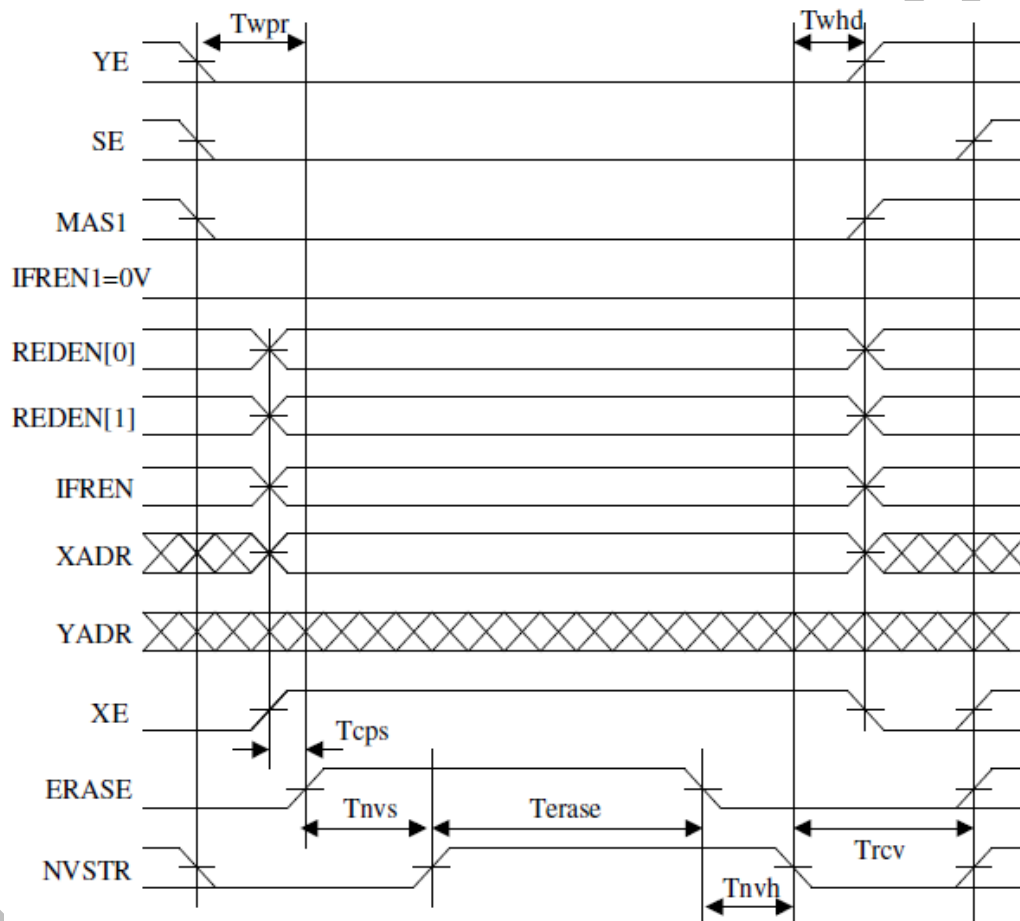


Рисунок 11. Временная диаграмма стирания страницы памяти

Запись 32-х битного слова в память.

Запись в память возможна только в режиме программирования. Для записи в память надо установить необходимое значение в бит IFREN (1 – для информационной памяти и 0 – для основной памяти), затем установить адрес, по которому производится запись, в регистре EEPROM_ADR, в регистр EEPROM_DI записать записываемое в память слово и установить биты XE и PROG в единицу, и спустя время $T_{nvs} = 5$ мкс установить бит NVSTR в единицу. Спустя время $T_{pgs} = 10$ мкс установить бит YE в единицу. Запись в память длится время $T_{prog} = 40$ мкс. Спустя это время необходимо очистить бит YE, и спустя время $T_{adh} = 20$ нс установить новый адрес и значение для записи в другую ячейку памяти. И спустя $T_{ads} = 20$ нс установить YE в единицу и записать следующую слово. Если запись больше не требуется, то спустя время $T_{pgh} = 20$ нс после очистки бита YE необходимо очистить бит PROG и спустя время $T_{nvh} = 5$ мкс очистить биты XE и NVSTR. Последующие операции с памятью можно выполнять спустя время $T_{rcv} = 10$ мкс. Временная диаграмма записи памяти представлена на рисунке.

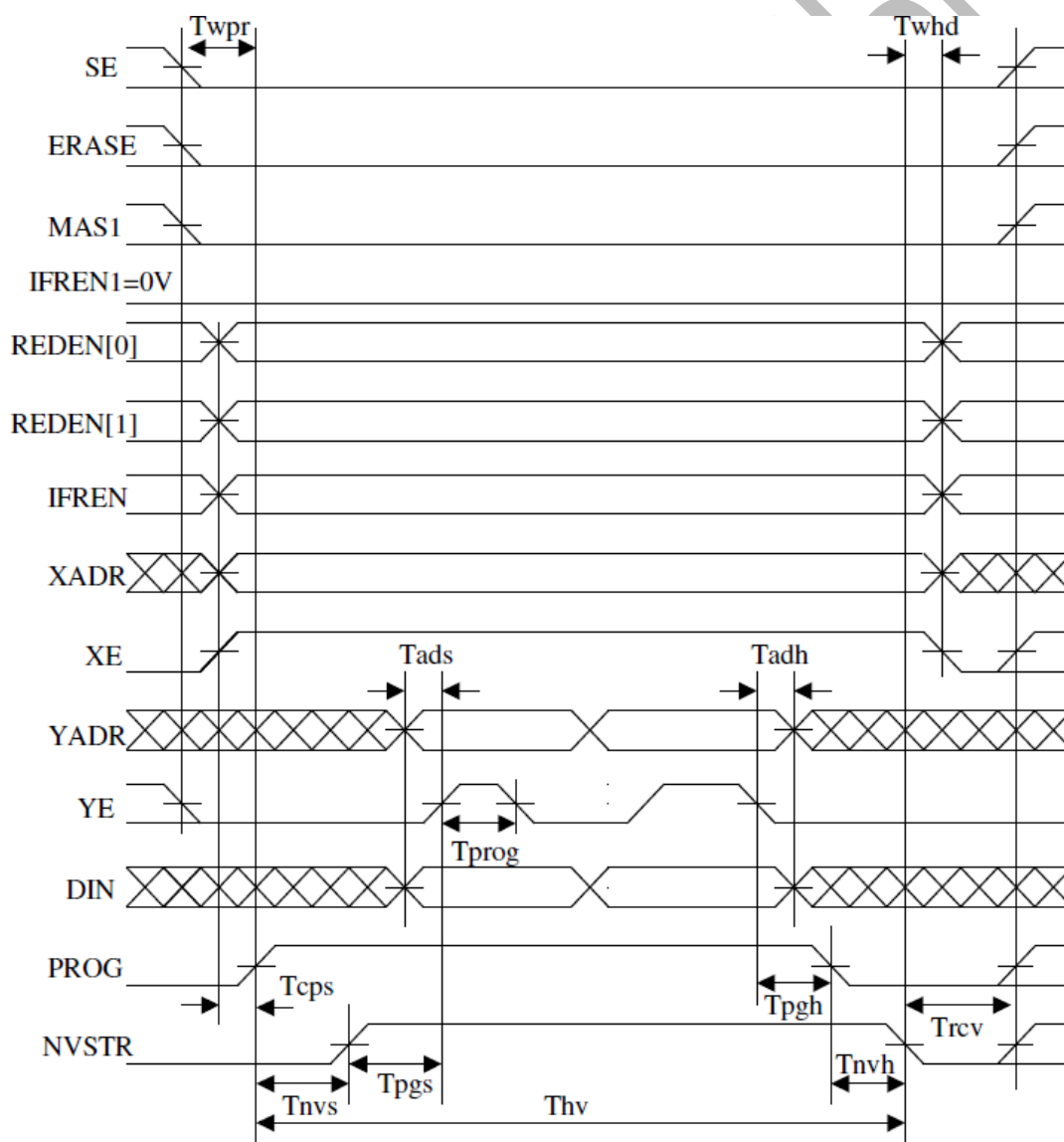


Рисунок 12. Временная диаграмма записи памяти

Чтение 32-х битного слова из памяти.

В обычном режиме работы для чтения доступна только основная память. Для этого необходимо просто считать требуемый адрес памяти. В режиме программирования для чтения доступна и основная и информационная память. Для чтения из памяти надо установить необходимое значение в бит IFREN (1 – для информационной памяти и 0 – для основной памяти), затем установить адрес, из которого необходимо считать данные в регистре EEPROM_ADR, и установить биты XE, YE и SE в единицу, и спустя время $T_{acc} = 30$ нс из регистра EEPROM_DO можно считать данные. Если необходимо считать следующее слово, то в регистр EEPROM_ADR необходимо записать новый адрес, сбросить бит SE на время $T_{nws} = 5$ нс, после этого установить SE и спустя время $T_{acc} = 30$ нс из регистра EEPROM_DO можно считать следующие данные. Если чтение больше не требуется, то можно очистить все биты управления. Временная диаграмма чтения памяти представлена на рисунке.

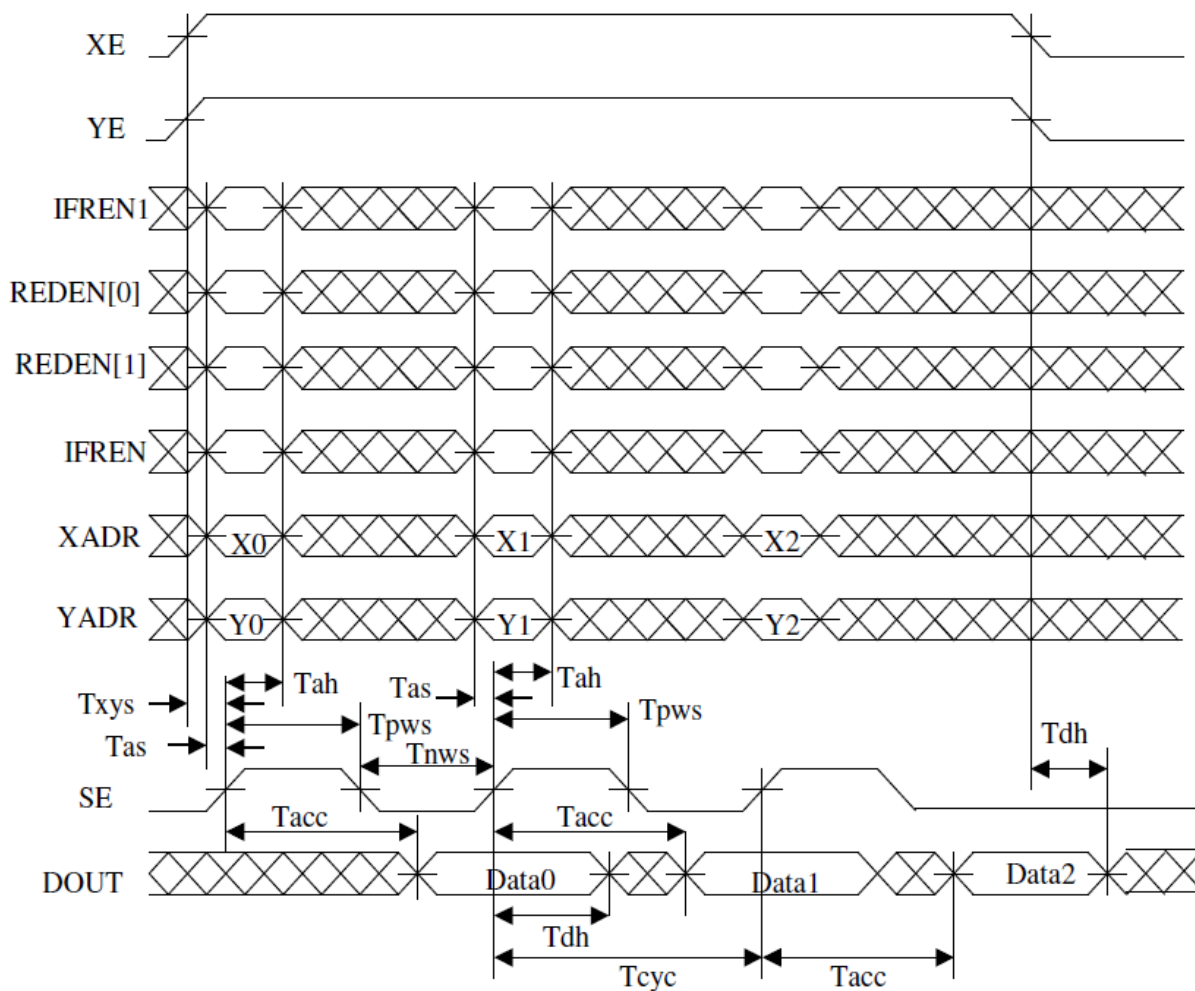


Рисунок 13. Временная диаграмма чтения памяти

Flash память программ поддерживает не менее 10000 циклов перезаписи. Нельзя повторять циклы стирания – записи, записи – записи и стирания – стирания одной ячейки памяти с периодом менее 16 мс.

Описание регистров управления контроллера Flash памяти программ

Таблица 12. Регистры управления контроллера Flash памяти программ

Базовый Адрес	Название	Описание
0x4001_8000	EEPROM_CNTRL	Регистры контроллера Flash памяти программ
Смещение		
0x00	EEPROM_CMD	Регистр управления EEPROM память
0x04	EEPROM_ADR	Регистр адреса
0x08	EEPROM_DI	Регистр данных на запись
0x0C	EEPROM_DO	Регистр данных считанных
0x10	EEPROM_KEY	Регистр ключа

EEPROM_CMD

Таблица 13. Регистр EEPROM_CMD

Номер	31...15	14	13	12	11	10	9	8
Доступ	U	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Сброс	0	1	0	0	0	0	0	0
		TMR	NVSTR	PROG	MAS1	ERASE	IFREN	SE

Номер	7	6	5...3	2...1	0
Доступ	R/W	R/W	R/W	U	R/W
Сброс	0	0	001	0	0
	YE	XE	Delay[2:0]		CON

R/W - бит доступен на чтение и запись

RO - бит доступен только на чтение

U - бит физически не реализован или зарезервирован.

Таблица 14. Описание бит регистра EEPROM_CMD

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...15	-	Зарезервировано
14	TMR	Сброс тестового режима flash: 0 – разрешение теста 1 – сброс Рекомендовано всегда записывать единицу.
13	NVSTR	Операции записи или стирания: 0 – при чтении; 1 – при записи или стирании
12	PROG	Записать данные по ADR[17:2] из регистра EEPROM_DI: 0 – нет записи; 1 – есть запись
11	MAS1	Стереть весь блок, при ERASE =1: 0 – нет стирания; 1 – стирание

10	ERASE	Стереть строку с адресом ADR[17:9], ADR[8:0] значения не имеет: 0 – нет стирания; 1 – стирание
9	IFREN	Работа с блоком информации: 0 – основная память; 1 – информационный блок
8	SE	Усилитель считывания: 0 – не включен; 1 – включен
7	YE	Выдача адреса ADR[8:2]: 0 – не разрешено; 1 – разрешено
6	XE	Выдача адреса ADR[17:9]: 0 – не разрешено; 1 – разрешено
5...3	Delay[2:0]	Задержка памяти программ при чтении в циклах (в рабочем режиме): 000 – 0 цикл 001 – 1 цикл
2...1	-	Зарезервировано
0	CON	Переключение контроллера памяти EEPROM на регистровое управление, не может производиться при исполнении программы из области EEPROM: 0 – управление EEPROM от ядра, рабочий режим; 1 – управление от регистров, режим программирования

EEPROM_ADR

Таблица 15. Регистр EEPROM_ADR

Номер	31...0
Доступ	R/W
Сброс	0
	ADR [31:0]

Таблица 16 Описание бит регистра EEPROM_ADR

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31..0	ADR[31:0]	Адрес обращения в память. ADR [1:0] – не имеет значения. Минимально адресуемая ячейка 32 бита

EEPROM_DI

Таблица 17. Регистр EEPROM_DI

Номер	31...0
Доступ	R/W
Сброс	0
	DATA [31:0]

Таблица 18. Описание бит регистра EEPROM_DI

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	DATA[31:0]	Данные для записи в EEPROM

EEPROM_DO

Таблица 19. Регистр EEPROM_DO

Номер	31...0
Доступ	R/W
Сброс	0
	DATA [31:0]

Таблица 20 Описание бит регистра EEPROM_DO

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	DATA[31:0]	Данные, считанные из EEPROM

EEPROM_KEY

Таблица 21. Регистр EEPROM_KEY

Номер	31...0
Доступ	R/W
Сброс	0
	KEY [31:0]

Таблица 22. Описание бит регистра EEPROM_KEY

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	KEY[31:0]	Ключ для разрешения доступа к Flash-памяти через регистровый доступ. Перед переводом памяти в режим программирования необходимо в регистр EEPROM_KEY записать комбинацию 0x8AAA5551. После окончания операций записи или стирания флеш необходимо сбросить ключ для запрещения регистрового доступа.

Система команд

В процессоре реализована RISC-V система команд RV32IMC.

Поддерживаемые команды представлены в Таблица 23.

В таблице используются следующие обозначения:

- rs – регистр источник;
- rd – регистр приёмник;
- imm или shamt – непосредственное значение;

Таблица 23. Система команд процессора BM-310S

Мнемокод команды	Формат	Операнды	Краткое описание
LB	I	rd, rs1, imm	Загрузка байта
LH	I	rd, rs1, imm	Загрузка полуслова
LW	I, Cx	rd, rs1, imm	Загрузка слова
LBU	I	rd, rs1, imm	Загрузка байта без знака
LHU	I	rd, rs1, imm	Загрузка слова без знака
SB	S	rs1, rs2, imm	Сохранение байта
SH	S	rs1, rs2, imm	Сохранение полуслова
SW	S, Cx	rs1, rs2, imm	Сохранение слова
ADD	R, Cx	rd, rs1, rs2	Арифметическое сложение
ADDI	I, Cx	rd, rs1, imm	Арифметическое сложение с непосредственным значением
SUB	R, Cx	rd, rs1, rs2	Арифметическое вычитание
LUI	U	rd, imm	Загрузка верхней части непосредственным значением
AUIPC	U	rd, imm	Арифметическое сложение верхней части РС с непосредственным значением
XOR	R	rd, rs1, rs2	Логическое исключающее “ИЛИ”
XORI	I	rd, rs1, imm	Логическое исключающее “ИЛИ” с непосредственным значением
OR	R, Cx	rd, rs1, rs2	Логическое “ИЛИ”
ORI	I	rd, rs1, imm	Логическое “ИЛИ” с непосредственным значением
AND	R, Cx	rd, rs1, rs2	Логическое “И”
ANDI	I	rd, rs1, imm	Логическое “И” с непосредственным значением
SLL	R	rd, rs1, rs2	Логический сдвиг влево
SLLI	I, Cx	rd, rs1, shamt	Логический сдвиг влево на непосредственное значение
SRL	R	rd, rs1, rs2	Логический сдвиг вправо
SRLI	I	rd, rs1, shamt	Логический сдвиг вправо на непосредственное значение
SRA	R	rd, rs1, rs2	Арифметический сдвиг вправо
SRAI	I	rd, rs1, shamt	Арифметический сдвиг вправо на непосредственное значение
SLT	R	rd, rs1, rs2	Сравнение “<”

Мнемокод команды	Формат	Операнды	Краткое описание
SLTI	I	rd, rs1, imm	Сравнение “<” с непосредственным значением
SLTU	R	rd, rs1, rs2	Сравнение “<” с беззнаковым
SLTIU	I	rd, rs1, imm	Сравнение “<” с беззнаковым с непосредственным значением
BEQ	SB, Cx	rs1, rs2, imm	Переход в случае равенства
BNE	SB, Cx	rs1, rs2, imm	Переход в случае неравенства
BLT	SB	rs1, rs2, imm	Переход в случае меньше
BGE	SB	rs1, rs2, imm	Переход в случае больше или равно
BLTU	SB	rs1, rs2, imm	Переход в случае меньше беззнакового
BGEU	SB	rs1, rs2, imm	Переход в случае больше или равно беззнакового
JAL	UJ, Cx	rd, imm	Переход & link
JALR	UJ, Cx	rd, rs1, imm	Переход & link register
FENCE.I	I	-	Синхронизация потока подкачки команд и потока чтения/записи данных
RDINSTRET	I	rd	Псевдоинструкция (alias) на команду чтения счетчика числа исполненных инструкций
ECALL	I	-	Запрос на выполнение переменного окружения операционной системы
EBREAK	I	-	Передача управления в переменное окружение отладки
RDCYCLE	I	rd	Псевдоинструкция (alias) на команду чтения счетчика количества выполненных циклов процессора
RDCYCLEH	I	rd	Псевдоинструкция (alias) на команду чтения старших бит счетчика количества выполненных циклов процессора
RDTIME	I	rd	Псевдоинструкция (alias) на команду чтения значения низкочастотного системного таймера ядра
RDTIMEH	I	rd	Псевдоинструкция (alias) на команду чтения значения старших бит низкочастотного системного таймера ядра
RDINSTRET	I	rd	Псевдоинструкция (alias) на команду чтения счетчика числа исполненных инструкций
RDINSTRETH	I	rd	Псевдоинструкция (alias) на команду чтения старших бит счетчика числа исполненных инструкций
MUL	R	rd, rs1, rs2	Умножение
MULH	R	rd, rs1, rs2	Умножение с возвратом старших бит результата

Мнемокод команды	Формат	Операнды	Краткое описание
MULHSU	R	rd, rs1, rs2	Умножение знакового на беззнаковое с возвратом старших бит результата
MULHU	R	rd, rs1, rs2	Умножение беззнакового на беззнаковое с возвратом старших бит результата
DIV	R	rd, rs1, rs2	Деление
DIVU	R	rd, rs1, rs2	Беззнаковое деление
REM	R	rd, rs1, rs2	Остаток от деления
REMU	R	rd, rs1, rs2	Беззнаковый остаток от деления

Подробное описание каждой команды и формат команд приведён в документе «The RISC-V Instruction Set Manual Volume I: User-Level ISA Document Version 2.2».

Процессорное ядро BM-310S

BM-310S поддерживает два режима привилегированности: machine и user. Режим user предоставляет механизм изоляции процессов друг от друга и от доверенного кода, исполняемого в machine режиме. Более подробно с режимами привилегированности можно ознакомиться в документе «The RISC-V Instruction Set Manual Volume II: Privileged Architecture».

Конвейер BM-310S состоит из 3-х стадий, на которых выполняются следующие операции:

1. Генерация запроса в подсистему памяти программ (PMS).
2. Чтение фрагмента кода из PMS и декодирование команды.
3. Исполнение команды.

Подсистема предварительной обработки команд (FE) организует выполнение операций первых двух стадий конвейера.

Блок FE состоит из следующих блоков:

1. IFU - Instruction Fetch Unit - содержит логику формирования адреса следующего фрагмента кода.
2. IDU - Instruction Decode Unit - блок декодирования инструкций.
3. RAS - Return Address Stack - стек адресов возврата вызовов функций.

На первой стадии блок IFU формирует запросы для считывания фрагментов кода в подсистему памяти программ (PMS), используя информацию о перенаправлении подкачки от IDU, RAS, а также подсистемы исполнения команд (BE).

На второй стадии происходит чтение фрагмента кода из памяти программ. Фрагмент кода может содержать одну или две команды. После этого осуществляется декодирование одной команды и результат помещается в очередь декодированных команд (состоит из 2-х элементов) или в ее байпас к подсистеме исполнения команд (BE).

На третьей стадии подсистема исполнения команд (BE) получает декодированную команду и ее операнды из регистрового файла, осуществляет проверку возможности ее выполнения и выполняет на соответствующем исполнительном устройстве.

Выполнение всех команд RV32IMC занимает 1 такт, кроме команд умножения/деления (таблица ниже).

Команда	Длительность, такты
MUL	2
MULH	2
MULHS	2
MULHSU	2
DIV	от 2 до 16
DIVU	от 2 до 16
REM	от 2 до 16
REMU	от 2 до 16

Исполнительное устройство умножения является конвейеризованным и отвечает за исполнение команд MUL/MULH[[S]U]. Исполнительное устройство деления является итеративным и отвечает за исполнение команд DIV[U] и REM[U]. Команды доступа к CSR регистрам исполняются в пустом конвейере.

Подсистемы памяти программ (PMS) и данных (DMS) обеспечивают доступ к интегрированной памяти (TCM) и I/O диапазону адресов, часть из которых зарезервирована

для внутренних блоков процессорного комплекса, а часть может быть использована для обращения к устройствам на АНВ шине. Доступ к ТСМ осуществляется с помощью выделенных интерфейсов. Ядро использует последовательную модель доступа (strong ordering memory access) по АНВ шине, т.е. следующий запрос на АНВ шине не будет выставлен, до окончания исполнения текущего. Если по окончании обработки запроса на АНВ шине возникла ошибка, то в ядре будет сгенерировано исключение (access fault).

Структурная схема процессора

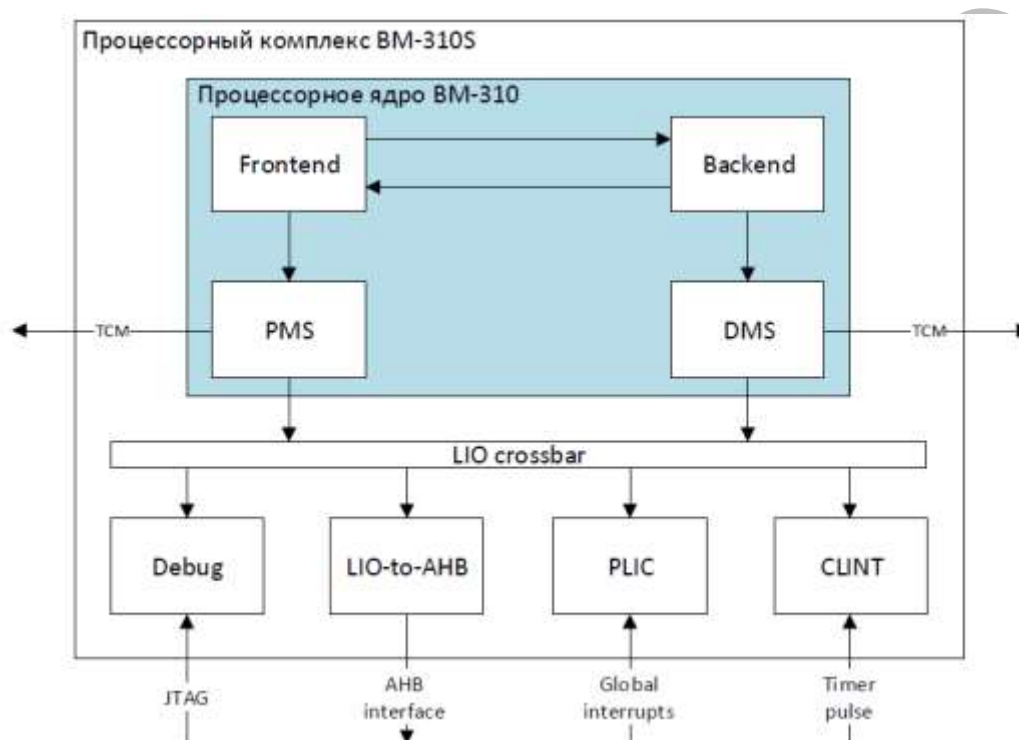


Рисунок 14. Структурная схема процессора

ТСМ интерфейс

ТСМ интерфейс используется для доступа к интегрированной памяти программ/данных. ТСМ интерфейс разделен на две части: интерфейс PMS и интерфейс DMS. Ширина шин адреса ТСМ интерфейса 14 бит, адресация пословная, поддерживается память размером до 128КБ. Ширина шин данных равна 32 бита.

АНВ I/O интерфейс

Процессорный комплекс использует АНВ I/O интерфейс для доступа к регистрам периферийных устройств или неэкшируемой памяти. АНВ I/O интерфейс использует 32-битный физический адрес. Ширина шин данных АНВ I/O интерфейса равна 32 бита. Существует возможность исполнения кода из диапазона I/O адресов.

Запросы прерываний

Запросы внешних прерываний Global interrupts подключены к контролеру внешних прерываний PLIC. Сигналы, подключаемые к Global interrupts, должны быть синхронны с тактовым сигналом ядра. Запросы прерываний являются edge-sensitive сигналами.

JTAG интерфейс

Имплементация должна устанавливать в корректное значение сигнал manufid - идентификатор производителя микросхемы (JEDEC ID). JTAG интерфейс подключается непосредственно к подсистеме отладки (Debug) процессорного комплекса. Подсистема отладки не сбрасывается от сигнала nRESET. Сброс блока Debug осуществляется сигналом POR. Для корректной работы блока Debug от должен быть сброшен при начале работы системы.

LIO crossbar

Локальная шина ядра.

Блок АЦП для измерения напряжений и токов в электрической сети

Микроконтроллер имеет в своем составе блок из 7 каналов 24 битных $\Sigma\Delta$ АЦП. Все каналы разбиты на три пары F0-F2 (канал напряжения и канал тока) для 3х фазной сети и еще одного независимого канала тока. Каждый из 7 каналов оцифровывает входной сигнал с выходной частотой отсчетов до 16 кГц. Кроме этого, в каждой паре каналов F0-F2 реализована возможность рассчитывать среднеквадратические значения тока/напряжения, вычислять активную и реактивную мощности, вычислять потребленную активную и реактивную энергию, частоту сигнала в каналах напряжения, превышение пикового значения, падение сигнала ниже установленного уровня. Эти дополнительные блоки позволяют снизить нагрузку на процессор, что в свою очередь снижает потребляемую мощность всего кристалла. Так же каждый АЦП имеет независимый канал DMA, обеспечивая возможность сохранения данных в ОЗУ без участия процессора. Необходимо учитывать, что запросы DMA от АЦП не маскируются, поэтому перед включением каналов АЦП требуется настройка соответствующих каналов DMA.

Структурная схема 7 каналов АЦП приведена ниже.

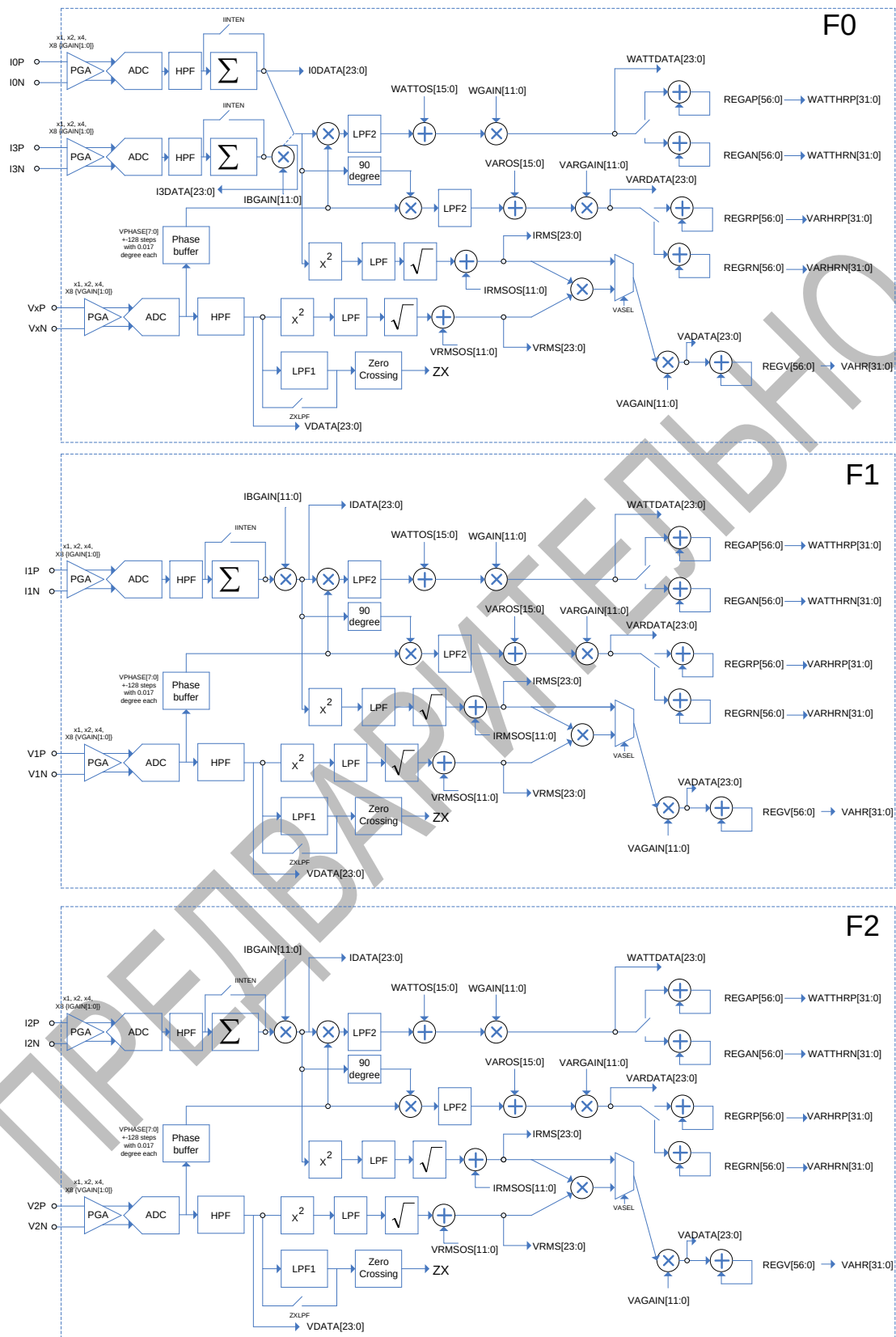


Рисунок 15. Структурная схема 7 каналов АЦП

Список вычисляемых параметров блоком АЦП:

- 7 независимых АЦП с выходной частотой отсчетов 4/8/16 кГц (4 канала тока и 3 канала напряжения). Эти каналы образуют 3 блока для измерения параметров каждой фазы F0-F2.
- В блоке каналов F0 реализуем автоматический выбор канала тока (который имеет максимальное значение) для последующих расчетов мощностных характеристик. Если разница токов превышает 6%, то формируется прерывание. Кроме этой функции в остальных блоках F0-F2 идентичны.
- Все каналы АЦП имеют независимые калибровочные коэффициенты наклона характеристики.
- Каждый канал тока имеет независимый интегратор.
- В каждом блоке АЦП (F0-F2) независимо рассчитывается период сигнала по каналу напряжения. Количество периодов, в течение которого рассчитывается эта величина, можно задавать равным 1/2/4/8/16/32/64/128 периодам.
- В каждом блоке есть проверка на пропажу периодического сигнала в канале напряжения.
- В каждом блоке проверяется просадка напряжения ниже заданного уровня, а также превышения сигнала в каналах тока и напряжения установленного лимита.
- Есть возможность скорректировать фазы сигналов в каналах напряжения с точностью до 0,02%.
- Вычисляются среднеквадратические, квадрат среднеквадратических значений токов и напряжений, а также их независимая калибровка.
- При вычислении активной и реактивной энергий значение накопленной энергии в течение периода накапливаются в отдельных регистрах (для положительной и отрицательной энергии).
- Вычисляются полная мощность и полная энергия.
- Вычисляется сдвиг фаз по отношению к фазе 0.

Для предотвращения влияния высокочастотных помех на результаты вычисления необходимо поставить внешний anti-aliasing фильтр. Можно использовать простейший RC фильтр первого порядка с частотой среза 100 кГц.

Все цифровые фильтры настроены на указанные частоты среза при входной частоте АЦП равной 1,024 МГц (выходной частоте отсчетов 4/8/16 кГц).

Описание регистров управления блока 7 каналов АЦП

Таблица 24 Регистры управления блока 7 каналов АЦП

Базовый Адрес	Название	Описание
0x4006_8000	ADCUI	Контроллер АЦП напряжения/тока
Смещение		
0x000	ADCUI_CTRL1	Общее управление для контроллера АЦП
0x004	ADCUI_CTRL2	
0x008	ADCUI_CTRL3	
0x00C	ADCUI_F0CTR	Управление в канале F0
0x010	ADCUI_F0WC	Управление расчета активной мощности в канале F0
0x014	ADCUI_F0WATTP	Старшая часть значения положительной активной мощности в канале F0
0x018	ADCUI_F0WATTN	Старшая часть значения отрицательной активной

		мощности в канале F0
0x01C	ADCUI_F0VC	Управление расчета реактивной мощности в канале F0
0x020	ADCUI_F0VARP	Старшая часть значения положительной реактивной мощности в канале F0
0x024	ADCUI_F0VARN	Старшая часть значения отрицательной реактивной мощности в канале F0
0x028	ADCUI_F0AC	Управление расчета полной мощности в канале F0
0x02C	ADCUI_F0VR	Старшая часть значения полной мощности в канале F0
0x030	ADCUI_F0MD0	Параметры 0 канала F0
0x034	ADCUI_F0MD1	Параметры 1 канала F0
0x038	ADCUI_F0VPEAK	Пиковое значение в канале напряжения в канале F0
0x03C	ADCUI_F0IPEAK	Пиковое значение в канале тока в канале F0
0x040	ADCUI_F0VDAT	Отсчеты напряжения в канале F0
0x044	ADCUI_F0I0DAT	Отсчеты тока I0 в канале F0
0x048	ADCUI_F0I3DAT	Отсчеты тока I3 в канале F0
0x04C	ADCUI_F0VRMS	Среднеквадратическое значение напряжение канала F0
0x050	ADCUI_F0VRMS2	Квадрат RMS в канале напряжения F0
0x054	ADCUI_F0IRMS	Среднеквадратическое значение тока канала F0
0x058	ADCUI_F0IRMS2	Квадрат RMS в канале тока F0
0x05C	ADCUI_F0STAT	Статус канала F0
0x060	ADCUI_F0MASK	Маска прерываний канала F0
0x064	ADCUI_F1CTR	Управление в канале F1
0x068	ADCUI_F1WC	Управление расчета активной мощности в канале F1
0x06C	ADCUI_F1WATTP	Старшая часть значения положительной активной мощности в канале F1
0x070	ADCUI_F1WATTN	Старшая часть значения отрицательной активной мощности в канале F1
0x074	ADCUI_F1VC	Управление расчета реактивной мощности в канале F1
0x078	ADCUI_F1VARP	Старшая часть значения положительной реактивной мощности в канале F1
0x07C	ADCUI_F1VARN	Старшая часть значения отрицательной реактивной мощности в канале F1
0x080	ADCUI_F1AC	Управление расчета полной мощности в канале F1
0x084	ADCUI_F1VR	Старшая часть значения полной мощности в канале F1
0x088	ADCUI_F1MD0	Параметры 0 канала F1
0x08C	ADCUI_F1MD1	Параметры 1 канала F1
0x090	ADCUI_F1MD2	Параметры 2 канала F1
0x094	ADCUI_F1VPEAK	Пиковое значение в канале напряжения в канале F1
0x098	ADCUI_F1IPEAK	Пиковое значение в канале тока в канале F1
0x09C	ADCUI_F1VDAT	Отсчеты напряжения в канале F1
0x0A0	ADCUI_F1IDAT	Отсчеты тока в канале F1
0x0A4	ADCUI_F1VRMS	Среднеквадратическое значение напряжение канала F1
0x0A8	ADCUI_F1VRMS2	Квадрат RMS в канале напряжения F1
0x0AC	ADCUI_F1IRMS	Среднеквадратическое значение тока канала F1
0x0B0	ADCUI_F1IRMS2	Квадрат RMS в канале тока F1
0x0B4	ADCUI_F1STAT	Статус канала F1
0x0B8	ADCUI_F1MASK	Маска прерываний канала F1

0x0BC	ADCUI_F2CTR	Управление в канале F2
0x0C0	ADCUI_F2WC	Управление расчета активной мощности в канале F2
0x0C4	ADCUI_F2WATTP	Старшая часть значения положительной активной мощности в канале F2
0x0C8	ADCUI_F2WATTN	Старшая часть значения отрицательной активной мощности в канале F2
0x0CC	ADCUI_F2VC	Управление расчета реактивной мощности в канале F2
0x0D0	ADCUI_F2VARP	Старшая часть значения положительной реактивной мощности в канале F2
0x0D4	ADCUI_F2VARN	Старшая часть значения отрицательной реактивной мощности в канале F2
0x0D8	ADCUI_F2AC	Управление расчета полной мощности в канале F2
0x0DC	ADCUI_F2VR	Старшая часть значения полной мощности в канале F2
0x0E0	ADCUI_F2MD0	Параметры 0 канала F2
0x0E4	ADCUI_F2MD1	Параметры 1 канала F2
0x0E8	ADCUI_F2MD2	Параметры 2 канала F2
0x0EC	ADCUI_F2VPEAK	Пиковое значение в канале напряжения в канале F2
0x0F0	ADCUI_F2IPEAK	Пиковое значение в канале тока в канале F2
0x0F4	ADCUI_F2VDAT	Отсчеты напряжения в канале F2
0x0F8	ADCUI_F2IDAT	Отсчеты тока в канале F2
0x0FC	ADCUI_F2VRMS	Среднеквадратическое значение напряжение канала F2
0x100	ADCUI_F2VRMS2	Квадрат RMS в канале напряжения F2
0x104	ADCUI_F2IRMS	Среднеквадратическое значение тока канала F2
0x108	ADCUI_F2IRMS2	Квадрат RMS в канале тока F2
0x10C	ADCUI_F2STAT	Статус канала F2
0x110	ADCUI_F2MASK	Маска прерываний канала F2
0x114	ADCUI_CCAL1	Регистр 1 калибровки канала тока
0x118	ADCUI_CCAL2	Регистр 2 калибровки канала тока
0x11C	ADCUI_CCAL3	Регистр 3 калибровки канала тока
0x120	ADCUI_CCAL4	Регистр 4 калибровки канала тока
0x124	ADCUI_F0IRMS_INACTIVE	Среднеквадратическое значение тока неактивного канала F0
0x128	ADCUI_F0WATTP_L	Младшая часть значения положительной активной мощности в канале F0
0x12C	ADCUI_F0WATTN_L	Младшая часть значения отрицательной активной мощности в канале F0
0x130	ADCUI_F0VARP_L	Младшая часть значения положительной реактивной мощности в канале F0
0x134	ADCUI_F0VARN_L	Младшая часть значения отрицательной реактивной мощности в канале F0
0x138	ADCUI_F0VR_L	Младшая часть значения полной мощности в канале F0
0x13C	ADCUI_F1WATTP_L	Младшая часть значения положительной активной мощности в канале F1
0x140	ADCUI_F1WATTN_L	Младшая часть значения отрицательной активной мощности в канале F1
0x144	ADCUI_F1VARP_L	Младшая часть значения положительной реактивной мощности в канале F1

0x148	ADCUI_F1VARN_L	Младшая часть значения отрицательной реактивной мощности в канале F1
0x14C	ADCUI_F1VR_L	Младшая часть значения полной мощности в канале F1
0x150	ADCUI_F2WATTP_L	Младшая часть значения положительной активной мощности в канале F2
0x154	ADCUI_F2WATTN_L	Младшая часть значения отрицательной активной мощности в канале F2
0x158	ADCUI_F2VARP_L	Младшая часть значения положительной реактивной мощности в канале F2
0x15C	ADCUI_F2VARN_L	Младшая часть значения отрицательной реактивной мощности в канале F2
0x160	ADCUI_F2VR_L	Младшая часть значения полной мощности в канале F2

ADCUI_CTRL1

Таблица 25 Регистр ADCUI_CTRL1

Номер	31...30	29	28	27	26...25	24:23
Доступ	R/W	R/W	R/W	R/W	-	R/W
Сброс	00	0	0	0	-	00
	OSR_CONF	tst_in[4]	RESET_DIG	ZXRMS	-	tst_in[3:2]

Номер	22	21	20	19	18...17	16...15
Доступ	R/W	R/W	R/W	R/W	R/W	R/W
Сброс	0	0	0	0	0	0
	tst_in[1]	tst_in[0]	VREF_SEL	FREQSEL	VANOLOAD	VARNLOAD

Номер	14	13...12	11...9	8	7	6
Доступ	-	R/W	R/W	R/W	R/W	R/W
Сброс	-	0	000	0	0	0
	-	APNOLOAD	PER_LENGTH	ZXLPF	RESOL	I3EN

Номер	5	4	3	2	1	0
Доступ	R/W	R/W	R/W	R/W	R/W	R/W
Сброс	0	0	0	0	0	0
	V2EN	I2EN	V1EN	I1EN	V0EN	I0EN

Таблица 26 Описание бит регистра ADCUI_CTRL1

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...30	OSR_CONF*	Выбор коэффициента передискретизации: 00 – 256 (4кГц); 01 – 128 (8кГц); 10 – 64 (16кГц); 11 – Зарезервировано
29	tst_in[4]	Тестовый выход (всегда записывать ноль)
28	RESET_DIG	Сброс цифровой части блоков АЦП: 0 – нет сброса; 1 – цифровая часть под общим сбросом
27	ZXRMS**	Управления обновления регистров со среднеквадратическими значениями: 0 – непрерывное обновление; 1 – обновление при пересечении напряжением “0”
26...25	-	Зарезервировано
24...23	tst_in[3:2]	Тестовые выходы (всегда записывать ноль)
22	tst_in[1]	Тестовый выход (всегда записывать ноль)
21	tst_in[0]	Тестовый выход (всегда записывать ноль)

20	VREF_SEL	Выбор опорного напряжения для АЦП: 0 – внешнее опорное напряжение; 1 – внутреннее опорное напряжение
19	FREQSEL	Разрешение вычисления длительности периода в каналах напряжения: 1 – разрешено; 0 – хранится последнее вычисленное значение
18..17	VANOLOAD	Режим “без нагрузки” при вычислении полной энергии: 00 – вся вычисленная энергия накапливается; 01 – не учитывается энергия ниже 0,012% от полной шкалы; 10 – не учитывается энергия ниже 0,0061% от полной шкалы; 11 – не учитывается энергия ниже 0,00305% от полной шкалы
16...1 5	VARNOLOAD	Режим “без нагрузки” при вычислении реактивной энергии: 00 – вся вычисленная энергия накапливается; 01 – не учитывается энергия ниже 0,012% от полной шкалы; 10 – не учитывается энергия ниже 0,0061% от полной шкалы; 11 – не учитывается энергия ниже 0,00305% от полной шкалы
14	-	Зарезервировано
13...1 2	APNOLOAD	Режим “без нагрузки” при вычислении активной энергии: 00 – вся вычисленная энергия накапливается; 01 – не учитывается энергия ниже 0,012% от полной шкалы; 10 – не учитывается энергия ниже 0,0061% от полной шкалы; 11 – не учитывается энергия ниже 0,00305% от полной шкалы
11...9	PER_LENGTH	Диапазон вычисления периода и фазового сдвига: 000 – в течение 1 периода; 001 – в течение 2 периодов; ... 111 – в течение 128 периодов
8	ZXLPF	Отключение низкочастотного фильтра перед детектором пересечения “0” в каналах напряжения: 0 – фильтр включен; 1 – фильтр отключен
7	RESOL	Разрешение выходных данных: 0 – 16 бит; 1 – 24 бита Старший бит в регистре данных 15 или 23. Сдвиг регистра данных выполнять не требуется.
6	I3EN	Разрешение работы канала I3: 0 – канал отключен; 1 – канал включен***
5	V2EN	Разрешение работы канала V2: 0 – канал отключен; 1 – канал включен***
4	I2EN	Разрешение работы канала I2: 0 – канал отключен; 1 – канал включен***

3	V1EN	Разрешение работы канала V1: 0 – канал отключен; 1 – канал включен***
2	I1EN	Разрешение работы канала I1: 0 – канал отключен; 1 – канал включен***
1	V0EN	Разрешение работы канала V0: 0 – канал отключен; 1 – канал включен***
0	I0EN	Разрешение работы канала I0: 0 – канал отключен; 1 – канал включен***

* - при увеличении частоты дискретизации все внутренние цифровые фильтры соответствующим образом корректируются, что сохраняет их частоты среза постоянными. Так же необходимо учитывать, что увеличение частоты дискретизации в 2 раза ведет к уменьшению THD+noise как минимум на 3 дБ в полосе от 0 Гц до половины частоты дискретизации (это следует из того, что шум интегрируется в частоте 2 раза большей).

** - так как происходит одновременное обновление среднеквадратических значений и тока и напряжение, то значение тока будет зависеть от угла между напряжением и током. На графике видна эта зависимость. Исходя из этих данных, можно скорректировать действительное значение тока.

***- при включении любого канала в блоке Fx, включаются FIFO всех остальных каналов (FIFO FxV DAT FxIDAT) в блоке Fx.

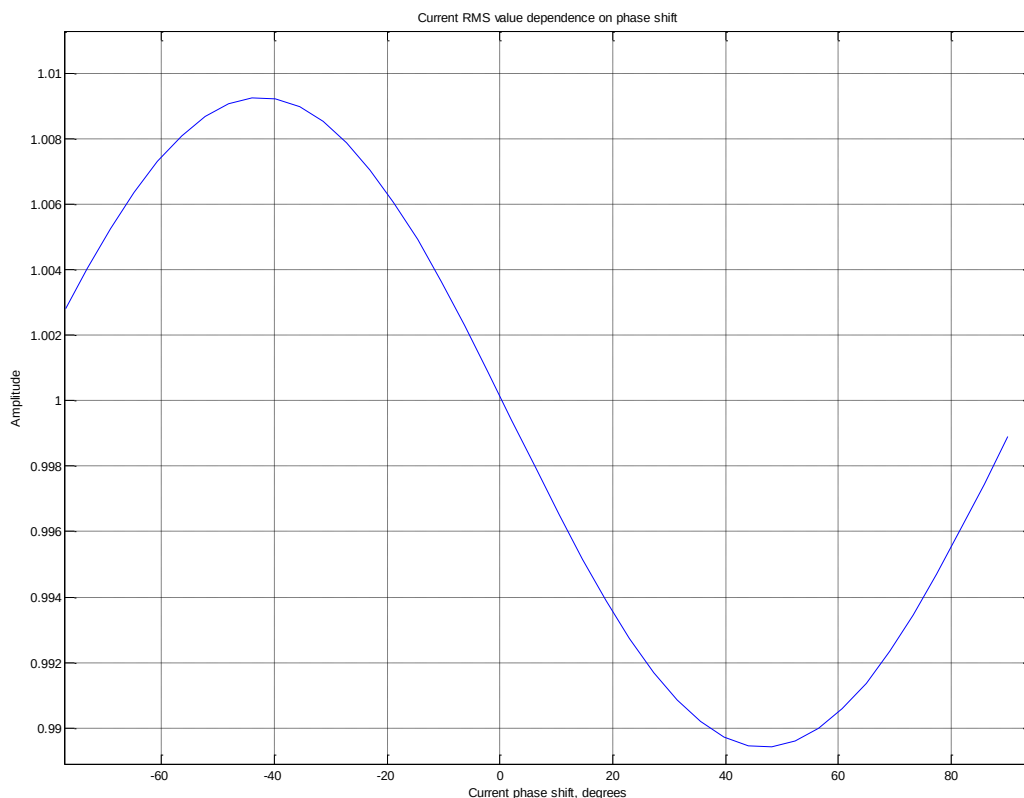


Рисунок 16. Зависимость RMS от сдвига фазы входного сигнала.

ПРЕДВАРИТЕЛЬНО

ADCUI_CTRL2

Таблица 27 Регистр ADCUI_CTRL2

Номер	31...24	23...16	15...0
Доступ	-	R/W	R/W
Сброс	-	FFh	FFFFh
	-	SAGCYC	SAGLVL

Таблица 28 Описание бит регистра ADCUI_CTRL2

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...24	-	Зарезервировано
23...16	SAGCYC	Количество полутактов напряжения для вычисления просадки уровня напряжения
15...0	SAGLVL	Уровень разрешенной просадки напряжения

ADCUI_CTRL3

Таблица 29 Регистр ADCUI_CTRL3

Номер	31...12	11...0
Доступ	-	R/W
Сброс	-	0FFh
	-	ZXTOUT

Таблица 30 Описание бит регистра ADCUI_CTRL3

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31..12	-	Зарезервировано
11..0	ZXTOUT	Значение time-out регистра, который устанавливает это значение при пересечении сигнала напряжения "0"

ADCUI_F0CTR

Таблица 31 Регистр ADCUI_F0CTR

Номер	31...20	19...18	17...10	9...8	7...6
Доступ	R/W	R/W	R/W	R/W	R/W
Сброс	0	00	00	0	0
	F0IRMSOS	F0I3GAIN	F0VPHASE	F0VGAIN	F0IGAIN

Номер	5	4	3	2	1	0
Доступ	WO	WO	WO	R/W	R/W	R/W
Сброс	0	0	0	0	0	0
	F0RVRS	F0RRRS	F0RARS	F0VASEL	F0I3NTEN	F0I0NTEN

Таблица 32 Описание бит регистра ADCUI_F0CTR

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...20	F0IRMSOS	Калибровка вычислителя среднеквадратического значения тока

19...18	F0I3GAIN	Предусилитель в канале тока 3: 00 – 0dB; 01 – 6dB; 10 – 12dB; 11 – 18dB
17...10	F0VPHASE	Фазовый сдвиг канала напряжения относительно канала тока, записанный в дополнительном коде. От -126d (-123мкс) до +127d(+124мкс). “0” соответствует синфазному сигналу с током.
9...8	F0VGAIN	Предусилитель в канале напряжения: 00 – 0dB; 01 – 6dB; 10 – 12dB; 11 – 18dB
7...6	F0I0GAIN	Предусилитель в канале тока 0: 00 – 0dB; 01 – 6dB; 10 – 12dB; 11 – 18dB
5	F0RVRS	Запись в этот регистр сбрасывает счетчик переданной полной энергии
4	F0RRRS	Запись в этот регистр сбрасывает счетчик переданной реактивной энергии
3	F0RARS	Запись в этот регистр сбрасывает счетчик переданной активной энергии
2	F0VASEL	Выбор источника сигнала для сохранения в регистре полной энергии: 0 – полная энергия; 1 – среднеквадратическое значение тока
1	F0I3NTEN	Отключение интегратора в канале тока 3: 0 – интегратор включен; 1 – интегратор отключен
0	F0I0NTEN	Отключение интегратора в канале тока 0: 0 – интегратор включен; 1 – интегратор отключен

ADCUI_F0WC

Таблица 33 Регистр ADCUI_F0WC

Номер	31...28	27...16	15...0
Доступ	-	R/W	R/W
Сброс	-	000h	0000h
	-	F0WGAIN	F0WATTOS

Таблица 34 Описание бит регистра ADCUI_F0WC

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...28	-	Зарезервировано
27...16	F0WGAIN	Калибровка усиления канала

15...0	F0WATTOS	Калибровка смещения канала
--------	----------	----------------------------

ПРЕДВАРИТЕЛЬНО

ADCUI_F0WATTP**Таблица 35 Регистр ADCUI_ F0WATTP**

Номер	31...0
Доступ	RO
Сброс	
	F0WATTHRP

Таблица 36 Описание бит регистра ADCUI_ F0WATTP

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	F0WATTHRP	Старшие 32 бита внутреннего 57 битного аккумулятора положительной активной энергии

ADCUI_F0WATTN**Таблица 37 Регистр ADCUI_ F0WATTN**

Номер	31...0
Доступ	RO
Сброс	
	F0WATTHRN

Таблица 38 Описание бит регистра ADCUI_ F0WATTN

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	F0WATTHRN	Старшие 32 бита внутреннего 57 битного аккумулятора отрицательной активной энергии

ADCUI_F0VC**Таблица 39 Регистр ADCUI_ F0VC**

Номер	31...28	27...16	15...0
Доступ	-	R/W	R/W
Сброс	-	000h	0000h
	-	F0VARGAIN	F0VAROS

Таблица 40 Описание бит регистра ADCUI_ F0VC

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...28	-	Зарезервировано
27...16	F0VARGAIN	Калибровка усиления канала
15...0	F0VAROS	Калибровка смещения канала

ADCUI_F0VARP**Таблица 41 Регистр ADCUI_ F0VARP**

Номер	31..0
Доступ	RO
Сброс	
	F0VARHRP

Таблица 42 Описание бит регистра ADCUI_F0VARP

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	F0VARHRP	Старшие 32 бита внутреннего 57-битного аккумулятора положительной реактивной энергии

ADCUI_F0VARN

Таблица 43 Регистр ADCUI_F0VARN

Номер	31..0
Доступ	RO
Сброс	
	F0VARHRN

Таблица 44 Описание бит регистра ADCUI_F0VARN

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	F0VARHRN	Старшие 32 бита внутреннего 57-битного аккумулятора отрицательной реактивной энергии

ADCUI_F0AC

Таблица 45 Регистр ADCUI_F0AC

Номер	31...28	27...16	15...13	12	11...0
Доступ	-	R/W	-	R/W	R/W
Сброс	-	000h	-	0	000h
	-	F0VAGAIN	-	F0I3SEL	F0VRMSOS

Таблица 46 Описание бит регистра ADCUI_F0AC

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...28	-	Зарезервировано
27...16	F0VAGAIN	Калибровка усиления канала
15...13	-	Зарезервировано
12	F0I3SEL	Выбор источника сигнала для регистра ADCUI_F0I3DAT: 0 – после фильтра высоких частот; 1 – до фильтра высоких частот
11...0	F0VRMSOS	Калибровка вычислителя среднеквадратического значения напряжения

ADCUI_F0VR

Таблица 47 Регистр ADCUI_F0VR

Номер	31...0
Доступ	RO
Сброс	
	F0VAHR

Таблица 48 Описание бит регистра ADCUI_F0VR

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	F0VAHR	Старшие 32 бита внутреннего 57-битного аккумулятора полной энергии

ADCUI_F0MD0

Таблица 49 Регистр ADCUI_F0MD0

Номер	31...30	29...9	8	7
Доступ	R/W	RO	R/W	R/W
Сброс	00		0	0
	F0SEL_I_CH	F0PER_FREQ	I3GAIN	V0GAIN

Номер	6	5	4	3:2	1:0
Доступ	R/W	R	R	R/W	R/W
Сброс	0	0	0	00	00
	I0GAIN	F0REACTS	F0ACTS	F0ISEL	F0VSEL

Таблица 50 Описание бит регистра ADCUI_F0MD0

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...30	F0SEL_I_CH	Выбор активного канала тока для вычисления мощностных характеристик: 00, 11 – автоматический выбор канала; 01 – активный канал I0; 10 – активный канал I3
29...9	F0PER_FREQ	Длительность такта в канале напряжения
8	I3GAIN	Усиление в канале I3: 0 – нет усиления; 1 – +6 дБ усиление
7	V0GAIN	Усиление в канале V0: 0 – нет усиления; 1 – +6 дБ усиление
6	I0GAIN	Усиление в канале I0: 0 – нет усиления; 1 – +6 дБ усиление
5	F0REACTS	Знак реактивной энергии в последний период
4	F0ACTS	Знак активной энергии в последний период
3...2	F0ISEL	Выбор источника сигнала для регистра ADCUI_F0IDAT: 00 – отсчеты тока; 01 – отсчеты активной мощности; 10 – отсчеты реактивной мощности; 11 – отсчеты полной мощности

1...0	F0VSEL	Выбор источника сигнала для регистра ADCUI_F0V DAT: 00 – отсчеты напряжения; 01 – отсчеты активной мощности; 10 – отсчеты реактивной мощности; 11 – отсчеты полной мощности
-------	--------	---

ПРЕДВАРИТЕЛЬНО

ADCUI_F0MD1

Таблица 51 Регистр ADCUI_ F0MD1

Номер	31...16	15...0
Доступ	R/W	R/W
Сброс	0000h	0000h
	F0VPKLVL	F0IPKLVL

Таблица 52 Описание бит регистра ADCUI_ F0MD1

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...16	F0VPKLVL	Предельный разрешенный уровень напряжения
15...0	F0IPKLVL	Предельный разрешенный уровень тока

ADCUI_F0VPEAK

Таблица 53 Регистр ADCUI_ F0VPEAK

Номер	31...24	23...0
Доступ		R/W
Сброс		000000h
	-	F0VPEAK

Таблица 54 Описание бит регистра ADCUI_ F0VPEAK

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...24	-	Зарезервировано
23...0	F0VPEAK	Пиковое значение напряжения. Обнуляется чтением из регистра Запись в этот регистр, сбрасывает пиковое значение

ADCUI_F0IPEAK

Таблица 55 Регистр ADCUI_ F0IPEAK

Номер	31:24	23:0
Доступ		R/W
Сброс		000000h
	-	F0IPEAK

Таблица 56 Описание бит регистра ADCUI_ F0IPEAK

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...24	-	Зарезервировано
23...0	F0IPEAK	Пиковое значение тока. Обнуляется чтением из регистра Запись в этот регистр, сбрасывает пиковое значение

ADCUI_F0VDAT

Таблица 57 Регистр ADCUI_ F0VDAT

Номер	31...24	23...0
Доступ		RO

Сброс		000000h
	-	F0V DAT

Таблица 58 Описание бит регистра ADCUI_ F0V DAT

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...24	-	Зарезервировано
23...0	F0V DAT	FIFO отсчетов напряжения (или одной из мощностей) FIFO включается при установке любого из бит I3EN, V0EN, I0EN

ADCUI_F0I0DAT

Таблица 59 Регистр ADCUI_ F0I0DAT

Номер	31...24	23...0
Доступ		RO
Сброс		000000h
	-	F0I0DAT

Таблица 60 Описание бит регистра ADCUI_ F0I0DAT

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...24	-	Зарезервировано
23...0	F0I0DAT	FIFO отсчетов тока 0 (или одной из мощностей) FIFO включается при установке любого из бит I3EN, V0EN, I0EN

ADCUI_F0I3DAT

Таблица 61 Регистр ADCUI_ F0I3DAT

Номер	31:24	23:0
Доступ		RO
Сброс		000000h
	-	F0I3DAT

Таблица 62 Описание бит регистра ADCUI_ F0I3DAT

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...24	-	Зарезервировано
23...0	F0I3DAT	FIFO отсчетов тока 3. FIFO включается при установке любого из бит I3EN, V0EN, I0EN

ADCUI_F0VRMS

Таблица 63 Регистр ADCUI_ F0VRMS

Номер	31...24	23...0
Доступ		RO
Сброс		000000h
	-	F0VRMS

Таблица 64 Описание бит регистра ADCUI_F0VRMS

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...24	-	Зарезервировано
23...0	F0VRMS	Среднеквадратическое значение напряжения

ADCUI_F0VRMS2

Таблица 65 Регистр ADCUI_F0VRMS2

Номер	31...0
Доступ	RO
Сброс	
	F0VRMS2

Таблица 66 Описание бит регистра ADCUI_F0VRMS2

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	F0VRMS2	Квадрат среднеквадратического значения напряжения

ADCUI_F0IRMS

Таблица 67 Регистр ADCUI_F0IRMS

Номер	31...24	23...0
Доступ		RO
Сброс		000000h
	-	F0IRMS

Таблица 68 Описание бит регистра ADCUI_F0IRMS

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...24	-	Зарезервировано
23...0	F0IRMS	Среднеквадратическое значение тока

ADCUI_F0IRMS2

Таблица 69 Регистр ADCUI_F0IRMS2

Номер	31...0
Доступ	RO
Сброс	
	F0IRMS2

Таблица 70 Описание бит регистра ADCUI_F0IRMS2

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	F0IRMS2	Квадрат среднеквадратического значения тока

ADCUI_F0STAT

Таблица 71 Регистр ADCUI_ F0STAT

Номер	31...27	26	25	24	23	22
Доступ	-	R/W	R/W	R/W	RO	RO
Сброс	-		0	0	0	0
	-	F0VAROVN	F0WATTOVN	C3IF_OVR	C3IF_FLL	C3IF_EMP

Номер	21	20	19	18	17	16
Доступ	R/W	RO	-	RO	R/W	RO
Сброс	0	0	-	0	0	0
	F0ZEROCRS	F0VANLDFL	-	F0VARNLDFL	F0VARSI GN	F0APNLDFL

Номер	15	14	13	12	11	10	9	8	7
Доступ	R/W	R/W	RO	R/W	R/W	R/W	R/W	R/W	R/W
Сброс	0	0	0	0	0	0	0	0	0
	F0APSI GN	F0FAU LTCON	F0ICHA NNEL	F0ZXT OF	F0VAO V	F0VAR OVP	F0WAT TOVP	F0PEA KIF	F0PEA KVF

Номер	6	5	4	3	2	1	0
Доступ	R/W	R/W	RO	RO	R/W	RO	RO
Сброс	0	0	0	0	0	0	0
	F0SAGF	F0IF_OVR	F0IF_FLL	F0IF_EMP	F0VF_OVR R	F0VF_FLL L	F0VF_EMP P

Таблица 72 Описание бит регистра ADCUI_ F0STAT

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...27	-	Зарезервировано
26	F0VAROVN	Флаг, что произошло переполнения регистра аккумулятора с отрицательной реактивной энергией. Запись “1” сбрасывает этот флаг
25	F0WATTOVN	Флаг, что произошло переполнения регистра аккумулятора с отрицательной активной энергией. Запись “1” сбрасывает этот флаг
24	C3IF_OVR	Флаг, что произошло переполнения FIFO F0I3DAT. Запись “1” сбрасывает этот флаг
23	C3IF_FLL	Флаг, что FIFO F0I3DAT заполнено
22	C3IF_EMP	Флаг, что FIFO F0I3DAT пусто
21	F0ZEROCRS	Флаг, что произошло пересечение “0” в канале напряжения. Запись “1” сбрасывает этот флаг
20	F0VANLDFL	Полная мощность ниже уровня сравнения
19	-	Зарезервировано

18	F0VARNLDFL	Реактивная мощность ниже уровня сравнения
17	F0VARSIGN	Смена знака реактивной мощности. Запись “1” сбрасывает этот флаг
16	F0APNLDFL	Активная мощность ниже уровня сравнения
15	F0APSIGN	Смена знака активной мощности. Запись “1” сбрасывает этот флаг
14	F0FAULTCON	Произошло автоматическое переключение активного канала тока. Запись “1” сбрасывает этот флаг
13	F0ICHANNEL	Активный канал тока: 0 – активный канал I0; 1 – активный канал I3
12	F0ZXTOF	Флаг, что в течении TimeOut не было пересечение напряжением значения “0”. Запись “1” сбрасывает этот флаг
11	F0VAOV	Флаг, что произошло переполнение регистра аккумулятора с полной энергией. Запись “1” сбрасывает этот флаг
10	F0VAROVP	Флаг, что произошло переполнение регистра аккумулятора с положительной реактивной энергией. Запись “1” сбрасывает этот флаг
9	F0WATTOVP	Флаг, что произошло переполнение регистра аккумулятора с положительной активной энергией. Запись “1” сбрасывает этот флаг
8	F0PEAKIF	Флаг, что произошло превышение порогового значения тока. Запись “1” сбрасывает этот флаг
7	F0PEAKVF	Флаг, что произошло превышение порогового значения напряжения. Запись “1” сбрасывает этот флаг
6	F0SAGF	Флаг, что произошла просадка напряжения. Запись “1” сбрасывает этот флаг
5	F0IF_OVR	Флаг, что произошло переполнение FIFO F0IDAT. Запись “1” сбрасывает этот флаг
4	F0IF_FLL	Флаг, что FIFO F0IDAT заполнено
3	F0IF_EMP	Флаг, что FIFO F0IDAT пусто
2	F0VF_OVR	Флаг, что произошло переполнение FIFO F0VDAT. Запись “1” сбрасывает этот флаг
1	F0VF_FLL	Флаг, что FIFO F0VDAT заполнено
0	F0VF_EMP	Флаг, что FIFO F0VDAT пусто

ADCUI_F0MASK

Таблица 73 Регистр ADCUI_F0MASK

Номер	31...27	26	25	24	23	22
Доступ	-	R/W	R/W	R/W	R/W	R/W
Сброс	-		0	0	0	0
	-	F0VAROV NM	F0WATTO VNM	C3IF_OVR M	C3IF_FLL M	C3IF_EMP M

Номер	21	20	19	18	17	16
Доступ	R/W	R/W	-	R/W	R/W	R/W
Сброс	0	0	-	0	0	0
	F0ZEROC RSM	F0VANLD FLM	-	F0VARNL DFLM	F0VARSIG NM	F0APNLD FLM

Номер	15	14	13	12	11	10	9	8	7
Доступ	R/W	R/W	-	R/W	R/W	R/W	R/W	R/W	R/W
Сброс	0	0	-	0	0	0	0	0	0
	F0APS IGNM	F0FA ULTC ONM	-	F0ZX TOFM	F0VA OVM	F0VA ROVP M	F0WA TTOV PM	F0PE AKIF M	F0PE AKVF M

Номер	6	5	4	3	2	1	0
Доступ	R/W	R/W	RO	RO	R/W	RO	RO
Сброс	0	0	0	0	0	0	0
	F0SAGF M	F0IF_OV RM	F0IF_FL LM	F0IF_EM PM	F0VF_O VRM	F0VF_FL LM	F0VF_E MPM

Таблица 74 Описание бит регистра ADCUI_ F0MASK

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...27	-	Зарезервировано
26	F0VAROVNM	Маска бита F0VAROVN
25	F0WATTOVNM	Маска бита F0WATTOVN
24	C3IF_OVRM	Маска бита C3IF_OVR
23	C3IF_FLLM	Маска бита C3IF_FLL
22	C3IF_EMPM	Маска бита C3IF_EMP
21	F0ZEROCRSM	Маска бита F0ZEROCRS
20	F0VANLDFLM	Маска бита F0VANLDLFL
19	-	Зарезервировано
18	F0VARNLDFLM	Маска бита F0VARNLDFL
17	F0VARSIGNM	Маска бита F0VARSIGN
16	F0APNLDFLM	Маска бита F0APNLDFL
15	F0APSIGNM	Маска бита F0APSIGNM
14	F0FAULTCONM	Маска бита F0FAULTCON
13	-	Зарезервировано
12	F0ZXTOFM	Маска бита F0ZXTOF
11	F0VAOVM	Маска бита F0VAOV
10	F0VAROVPM	Маска бита F0VAROVPM
9	F0WATTOVPM	Маска бита F0WATTOVPM
8	F0PEAKIFM	Маска бита F0PEAKIF
7	F0PEAKVFM	Маска бита F0PEAKVF
6	F0SAGFM	Маска бита F0SAGF

5	F0IF_OVRM	Маска бита F0IF_OVR
4	F0IF_FLLM	Маска бита F0IF_FLL
3	F0IF_EMPM	Маска бита F0IF_EMP
2	F0VF_OVRM	Маска бита F0VF_OVR
1	F0VF_FLLM	Маска бита F0VF_FLL
0	F0VF_EMPM	Маска бита F0VF_EMP

ПРЕДВАРИТЕЛЬНО

ADCUI_F1CTR

Таблица 75 Регистр ADCUI_ F1CTR

Номер	31:20	19:18	17:10	9:8	7:6
Доступ	R/W	-	R/W	R/W	R/W
Сброс	0	-	00	0	0
	F1IRMSOS	-	F1VPHASE	F1VGAIN	F1IGAIN

Номер	5	4	3	2	1	0
Доступ	WO	WO	WO	R/W	-	R/W
Сброс	0	0	0	0	-	0
	F1RVRS	F1RRRS	F1RARS	F1VASEL	-	F1INTEN

Таблица 76 Описание бит регистра ADCUI_ F1CTR

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...20	F1IRMSOS	Калибровка вычислителя среднеквадратического значения тока
19...18	-	Зарезервировано
17...10	F1VPHASE	Фазовый сдвиг канала напряжения относительно канала тока, записанный в дополнительном коде. От -126d (-123мкс) до +127d(+124мкс). "0" соответствует синфазному сигналу с током.
9...8	F1VGAIN	Предусилитель в канале напряжения: 00 – 0dB; 01 – 6dB; 10 – 12dB; 11 – 18dB
7...6	F1IGAIN	Предусилитель в канале тока: 00 – 0dB; 01 – 6dB; 10 – 12dB; 11 – 18dB
5	F1RVRS	Запись в этот регистр сбрасывает счетчик переданной полной энергии
4	F1RRRS	Запись в этот регистр сбрасывает счетчик переданной реактивной энергии
3	F1RARS	Запись в этот регистр сбрасывает счетчик переданной активной энергии
2	F1VASEL	Выбор источника сигнала для сохранения в регистре полной энергии: 0 – полная энергия; 1 – среднеквадратическое значение тока
1	-	Зарезервировано
0	F1INTEN	Отключение интегратора в канале тока: 0 – интегратор включен; 1 – интегратор отключен

ПРЕДВАРИТЕЛЬНО

ADCUI_F1WC

Таблица 77 Регистр ADCUI_F1WC

Номер	31...28	27...16	15...0
Доступ	-	R/W	R/W
Сброс	-	000h	0000h
	-	F1WGAIN	F1WATTOS

Таблица 78 Описание бит регистра ADCUI_F1WC

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...28	-	Зарезервировано
27...16	F1WGAIN	Калибровка усиления канала
15...0	F1WATTOS	Калибровка смещения канала

ADCUI_F1WATTP

Таблица 79 Регистр ADCUI_F1WATTP

Номер	31...0
Доступ	RO
Сброс	
	F1WATTHRP

Таблица 80 Описание бит регистра ADCUI_F1WATTP

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	F1WATTHRP	Старшие 32 бита внутреннего 57-битного аккумулятора положительной активной энергии

ADCUI_F1WATTN

Таблица 81 Регистр ADCUI_F1WATTN

Номер	31...0
Доступ	RO
Сброс	
	F1WATTHRN

Таблица 82 Описание бит регистра ADCUI_F1WATTN

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	F1WATTHRN	Старшие 32 бита внутреннего 57-битного аккумулятора отрицательной активной энергии

ADCUI_F1VC

Таблица 83 Регистр ADCUI_F1VC

Номер	31...28	27...16	15...0
Доступ	-	R/W	R/W
Сброс	-	000h	0000h
	-	F1VARGAIN	F1VAROS

Таблица 84 Описание бит регистра ADCUI_ F1VC

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...28	-	Зарезервировано
27...16	F1VARGAIN	Калибровка усиления канала
15...0	F1VAROS	Калибровка смещения канала

ADCUI_F1VARP

Таблица 85 Регистр ADCUI_ F1VARP

Номер	31...0
Доступ	RO
Сброс	
	F1VARHRP

Таблица 86 Описание бит регистра ADCUI_ F1VARP

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	F1VARHRP	Старшие 32 бита внутреннего 57-битного аккумулятора положительной реактивной энергии

ADCUI_F1VARN

Таблица 87 Регистр ADCUI_ F1VARN

Номер	31...0
Доступ	RO
Сброс	
	F1VARHRN

Таблица 88 Описание бит регистра ADCUI_ F1VARN

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	F1VARHRN	Старшие 32 бита внутреннего 57-битного аккумулятора отрицательной реактивной энергии

ADCUI_F1AC

Таблица 89 Регистр ADCUI_ F1AC

Номер	31...28	27...16	15...12	11...0
Доступ	-	R/W	-	R/W
Сброс	-	000h	-	000h
	-	F1VAGAIN	-	F1VRMSOS

Таблица 90 Описание бит регистра ADCUI_ F1AC

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...28	-	Зарезервировано
27...16	F1VAGAIN	Калибровка усиления канала
15...12	-	Зарезервировано

11...0	F1VRMSOS	Калибровка вычислителя среднеквадратического значения напряжения
--------	----------	--

ПРЕДВАРИТЕЛЬНО

ADCUI_F1VR

Таблица 91 Регистр ADCUI_ F1VR

Номер	31...0
Доступ	RO
Сброс	
	F1VAHR

Таблица 92 Описание бит регистра ADCUI_ F1VR

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	F1VAHR	Старшие 32 бита внутреннего 57-битного аккумулятора полной энергии

ADCUI_F1MD0

Таблица 93 Регистр ADCUI_ F1MD0

Номер	31..29	28..8	7	6	5	4	3..2	1..0
Доступ	-	RO	R/W	R/W	R	R	R/W	R/W
Сброс	-		0	0	0	0	00	00
	-	F1PER_FREQ	V1GAIN	I1GAIN	F1REACTS	F1ACTS	F1ISEL	F1VSEL

Таблица 94 Описание бит регистра ADCUI_ F1MD0

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...29	-	Зарезервировано
28...8	F1PER_FREQ	Длительность такта в канале напряжения
7	V1GAIN	Усиление в канале V1: 0 – нет усиления; 1 – +6 дБ усиление
6	I1GAIN	Усиление в канале I1: 0 – нет усиления; 1 – +6 дБ усиление
5	F1REACTS	Знак реактивной энергии в последний период
4	F1ACTS	Знак активной энергии в последний период
3...2	F1ISEL	Выбор источника сигнала для регистра ADCUI_F1IDAT: 00 – отсчеты тока; 01 – отсчеты активной мощности; 10 – отсчеты реактивной мощности; 11 – отсчеты полной мощности
1...0	F1VSEL	Выбор источника сигнала для регистра ADCUI_F1VDAT: 00 – отсчеты напряжения; 01 – отсчеты активной мощности; 10 – отсчеты реактивной мощности; 11 – отсчеты полной мощности

ADCUI_F1MD1**Таблица 95 Регистр ADCUI_ F1MD1**

Номер	31...16	15...0
Доступ	R/W	R/W
Сброс	0000h	0000h
	F1VPKLVL	F1IPKLVL

Таблица 96 Описание бит регистра ADCUI_ F1MD1

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...16	F1VPKLVL	Предельный разрешенный уровень напряжения
15...0	F1IPKLVL	Предельный разрешенный уровень тока

ADCUI_F1MD2**Таблица 97 Регистр ADCUI_ F1MD2**

Номер	31...17	16...0
Доступ	-	RO
Сброс	-	00000h
	-	F1PHASE

Таблица 98 Описание бит регистра ADCUI_ F1MD2

№	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...17	-	Зарезервировано
16...0	F1PHASE	Фазовый сдвиг канала напряжения V1 по отношению к V0

ADCUI_F1VPEAK**Таблица 99 Регистр ADCUI_ F1VPEAK**

Номер	31...24	23...0
Доступ		R/W
Сброс		000000h
	-	F1VPEAK

Таблица 100 Описание бит регистра ADCUI_ F1VPEAK

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...24	-	Зарезервировано
23...0	F1VPEAK	Пиковое значение напряжения. Обнуляется чтением из регистра Запись в этот регистр, сбрасывает пиковое значение

ADCUI_F1IPEAK**Таблица 101 Регистр ADCUI_ F1IPEAK**

Номер	31...24	23...0
Доступ		R/W
Сброс		000000h

	-	F1IPEAK
--	---	----------------

Таблица 102 Описание бит регистра ADCUI_ F1IPEAK

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...24	-	Зарезервировано
23...0	F1IPEAK	Пиковое значение тока. Обнуляется чтением из регистра Запись в этот регистр, сбрасывает пиковое значение

ADCUI_F1VDAT

Таблица 103 Регистр ADCUI_ F1VDAT

Номер	31...24	23...0
Доступ		RO
Сброс		000000h
	-	F1VDAT

Таблица 104 Описание бит регистра ADCUI_ F1VDAT

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...24	-	Зарезервировано
23...0	F1VDAT	FIFO отсчетов напряжения (или одной из мощностей). FIFO включается при установке любого из бит V1EN, I1EN

ADCUI_F1IDAT

Таблица 105 Регистр ADCUI_ F1IDAT

Номер	31...24	23...0
Доступ		RO
Сброс		000000h
	-	F1IDAT

Таблица 106 Описание бит регистра ADCUI_ F1IDAT

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...24	-	Зарезервировано
23...0	F1IDAT	FIFO отсчетов тока (или одной из мощностей). FIFO включается при установке любого из бит V1EN, I1EN

ADCUI_F1VRMS

Таблица 107 Регистр ADCUI_ F1VRMS

Номер	31...24	23...0
Доступ		RO
Сброс		000000h
	-	F1VRMS

Таблица 108 Описание бит регистра ADCUI_ F1VRMS

Разряды	Функциональное	Расшифровка функционального имени бита, краткое
---------	----------------	---

	имя бита	описание назначения и принимаемых значений
31...24	-	Зарезервировано
23..0	F1VRMS	Среднеквадратическое значение напряжения

ADCUI_F1VRMS2

Таблица 109 Регистр ADCUI_ F1VRMS2

Номер	31...0
Доступ	RO
Сброс	
	F1VRMS2

Таблица 110 Описание бит регистра ADCUI_ F1VRMS2

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	F1VRMS2	Квадрат среднеквадратического значения напряжения

ADCUI_F1IRMS

Таблица 111 Регистр ADCUI_ F1IRMS

Номер	31...24	23...0
Доступ		R/W
Сброс		000000h
	-	F1IRMS

Таблица 112 Описание бит регистра ADCUI_ F1IRMS

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...24	-	Зарезервировано
23...0	F1IRMS	Среднеквадратическое значение тока

ADCUI_F1IRMS2

Таблица 113 Регистр ADCUI_ F1IRMS2

Номер	31...0
Доступ	RO
Сброс	
	F1IRMS2

Таблица 114 Описание бит регистра ADCUI_ F1IRMS2

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	F1IRMS2	Квадрат среднеквадратического значения тока

ADCUI_F1STAT

Таблица 115 Регистр ADCUI_ F1STAT

Номер	31...27	26	25	24...22	21
Доступ	-	R/W	R/W	-	R/W
Сброс	-		0	-	0

	-	F1VAROVN	F1WATTOVN	-	F1ZERO CRS
--	---	-----------------	------------------	---	-------------------

Номер	20	19	18	17	16	15
Доступ	RO	-	RO	R/W	RO	R/W
Сброс	0	-	0	0	0	0
	F1VANLDFL	-	F1VARNLDFL	F1VARSIGN	F1APNLDFL	F1APSIGN

Номер	14...13	12	11	10	9	8	7
Доступ	-	R/W	R/W	R/W	R/W	R/W	R/W
Сброс	-	0	0	0	0	0	0
	-	F1ZXTOF	F1VAOV	F1VAROVP	F1WATTOVP	F1PEAKIF	F1PEAKVF

Номер	6	5	4	3	2	1	0
Доступ	R/W	R/W	RO	RO	R/W	RO	RO
Сброс	0	0	0	0	0	0	0
	F1SAGF	F1IF_OVR	F1IF_FL	F1IF_EMP	F1VF_OVR	F1VF_FL	F1VF_EMP

Таблица 116 Описание бит регистра ADCUI_ F1STAT

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...27	-	Зарезервировано.
26	F1VAROVN	Флаг, что произошло переполнения регистра аккумулятора с отрицательной реактивной энергией. Запись "1" сбрасывает этот флаг
25	F1WATTOVN	Флаг, что произошло переполнения регистра аккумулятора с отрицательной активной энергией. Запись "1" сбрасывает этот флаг
24...22	-	Зарезервировано
21	F1ZERO CRS	Флаг, что произошло пересечение "0" в канале напряжения. Запись "1" сбрасывает этот флаг
20	F1VANLDFL	Полная мощность ниже уровня сравнения
19	-	Зарезервировано
18	F1VARNLDFL	Реактивная мощность ниже уровня сравнения
17	F1VARSIGN	Смена знака реактивной мощности. Запись "1" сбрасывает этот флаг
16	F1APNLDFL	Активная мощность ниже уровня сравнения.
15	F1APSIGN	Смена знака активной мощности. Запись "1" сбрасывает этот флаг
14...13	-	Зарезервировано
12	F1ZXTOF	Флаг, что в течении TimeOut не было пересечение напряжением значения "0". Запись "1" сбрасывает этот флаг
11	F1VAOV	Флаг, что произошло переполнения регистра аккумулятора с полной энергией. Запись "1" сбрасывает этот флаг

10	F1VAROVP	Флаг, что произошло переполнения регистра аккумулятора с положительной реактивной энергией. Запись “1” сбрасывает этот флаг
9	F1WATTOVP	Флаг, что произошло переполнения регистра аккумулятора с положительной активной энергией. Запись “1” сбрасывает этот флаг.
8	F1PEAKIF	Флаг, что произошло превышение порогового значения тока. Запись “1” сбрасывает этот флаг
7	F1PEAKVF	Флаг, что произошло превышение порогового значения напряжения. Запись “1” сбрасывает этот флаг
6	F1SAGF	Флаг, что произошла просадка напряжения. Запись “1” сбрасывает этот флаг
5	F1IF_OVR	Флаг, что произошло переполнения FIFO F1IDAT. Запись “1” сбрасывает этот флаг
4	F1IF_FLL	Флаг, что FIFO F1IDAT заполнено
3	F1IF_EMP	Флаг, что FIFO F1IDAT пусто
2	F1VF_OVR	Флаг, что произошло переполнения FIFO F1VDAT. Запись “1” сбрасывает этот флаг
1	F1VF_FLL	Флаг, что FIFO F1VDAT заполнено
0	F1VF_EMP	Флаг, что FIFO F1VDAT пусто

ADCUI_F1MASK

Таблица 117 Регистр ADCUI_F1MASK

Номер	31...27	26	25	24...22	21
Доступ	-	R/W	R/W	-	R/W
Сброс	-		0	-	0
	-	F1VAROVNM	F1WATTOVNM	-	F1ZEROCRSM

Номер	20	19	18	17	16
Доступ	R/W	-	R/W	R/W	R/W
Сброс	0	-	0	0	0
	F1VANLDFLM	-	F1VARNLDFLM	F1VARSIGNM	F1APNLDFLM

Номер	15	14...13	12	11	10	9	8	7
Доступ	R/W	-	R/W	R/W	R/W	R/W	R/W	R/W
Сброс	0	-	0	0	0	0	0	0
	F1APSIGNM	-	F1ZXTOFM	F1VAOVM	F1VAROVPM	F1WATTOVPM	F1PEAKIFM	F1PEAKVFM

Номер	6	5	4	3	2	1	0
Доступ	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Сброс	0	0	0	0	0	0	0
	F1SAGFM	F1IF_OVRM	F1IF_FLLM	F1IF_EMPM	F1VF_OVRM	F1VF_FLLM	F1VF_EMPM

Таблица 118 Описание бит регистра ADCUI_ F1MASK

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...27	-	Зарезервировано
26	F1VAROVNM	Маска бита F1VAROVN
25	F1WATTOVNM	Маска бита F1WATTOVN
24...22	-	Зарезервировано
21	F1ZEROCRSM	Маска бита F1ZEROCRS
20	F1VANLDFLM	Маска бита F1VANLDFL
19	-	Зарезервировано
18	F1VARNLDFLM	Маска бита F1VARNLDFL
17	F1VARSIGNM	Маска бита F1VARSIGN
16	F1APNLDFLM	Маска бита F1APNLDFL
15	F1APSIGNM	Маска бита F1APSIGNM
14...13	-	Зарезервировано
12	F1ZXTOFM	Маска бита F1ZXTOF
11	F1VAOVM	Маска бита F1VAOV
10	F1VAROVPM	Маска бита F1VAROVP
9	F1WATTOVPM	Маска бита F1WATTOVP
8	F1PEAKIFM	Маска бита F1PEAKIF
7	F1PEAKVFM	Маска бита F1PEAKVF
6	F1SAGFM	Маска бита F1SAGF
5	F1IF_OVRM	Маска бита F1IF_OVR
4	F1IF_FLLM	Маска бита F1IF_FLL
3	F1IF_EMPM	Маска бита F1IF_EMP
2	F1VF_OVRM	Маска бита F1VF_OVR
1	F1VF_FLLM	Маска бита F1VF_FLL
0	F1VF_EMPM	Маска бита F1VF_EMP

ADCUI_F2CTR

Таблица 119 Регистр ADCUI_ F2CTR

Номер	31...20	19...18	17...10	9...8	7...6
Доступ	R/W	-	R/W	R/W	R/W
Сброс	0	-	00	0	0
	F2IRMSOS	-	F2VPHASE	F2VGAIN	F2IGAIN

Номер	5	4	3	2	1	0
Доступ	WO	WO	WO	R/W	-	R/W
Сброс	0	0	0	0	-	0
	F2RVRS	F2RRRS	F2RARS	F2VASEL	-	F2INTEN

Таблица 120 Описание бит регистра ADCUI_ F2CTR

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...20	F2IRMSOS	Калибровка вычислителя среднеквадратического значения тока

19...18	-	Зарезервировано
17..10	F2VPHASE	Фазовый сдвиг канала напряжения относительно канала тока, записанный в дополнительном коде. От -126d (-123мкс) до +127d(+124мкс). "0" соответствует синфазному сигналу с током.
9...8	F2VGAIN	Предусилитель в канале напряжения: 00 – 0dB; 01 – 6dB; 10 – 12dB; 11 – 18dB
7...6	F2IGAIN	Предусилитель в канале тока: 00 – 0dB; 01 – 6dB; 10 – 12dB; 11 – 18dB
5	F2RVRS	Запись в этот регистр сбрасывает счетчик переданной полной энергии
4	F2RRRS	Запись в этот регистр сбрасывает счетчик переданной реактивной энергии
3	F2RARS	Запись в этот регистр сбрасывает счетчик переданной активной энергии
2	F2VASEL	Выбор источника сигнала для сохранения в регистре полной энергии: 0 – полная энергия; 1 – среднеквадратическое значение тока
1	-	Зарезервировано
0	F2INTEN	Отключение интегратора в канале тока: 0 – интегратор включен; 1 – интегратор отключен

ADCUI_F2WC

Таблица 121 Регистр ADCUI_F2WC

Номер	31...28	27...16	15...0
Доступ	-	R/W	R/W
Сброс	-	000h	0000h
	-	F2WGAIN	F2WATTOS

Таблица 122 Описание бит регистра ADCUI_F2WC

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...28	-	Зарезервировано
27...16	F2WGAIN	Калибровка усиления канала
15...0	F2WATTOS	Калибровка смещения канала

ADCUI_F2WATTP

Таблица 123 Регистр ADCUI_F2WATTP

Номер	31...0
-------	--------

Доступ	RO
Сброс	
	F2WATTHRP

Таблица 124 Описание бит регистра ADCUI_ F2WATTP

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	F2WATTHRP	Старшие 32 бита внутреннего 57-битного аккумулятора положительной активной энергии

ADCUI_F2WATTN

Таблица 125 Регистр ADCUI_ F2WATTN

Номер	31...0
Доступ	RO
Сброс	
	F2WATTHRN

Таблица 126 Описание бит регистра ADCUI_ F2WATTN

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	F2WATTHRN	Старшие 32 бита внутреннего 57-битного аккумулятора отрицательной активной энергии

ADCUI_F2VC

Таблица 127 Регистр ADCUI_ F2VC

Номер	31...28	27...16	15...0
Доступ	-	R/W	R/W
Сброс	-	000h	0000h
	-	F2VARGAIN	F2VAROS

Таблица 128 Описание бит регистра ADCUI_ F2VC

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...28	-	Зарезервировано
27...16	F2VARGAIN	Калибровка усиления канала
15...0	F2VAROS	Калибровка смещения канала

ADCUI_F2VARP

Таблица 129 Регистр ADCUI_ F2VARP

Номер	31...0
Доступ	RO
Сброс	
	F2VARHRP

Таблица 130 Описание бит регистра ADCUI_ F2VARP

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	F2VARHRP	Старшие 32 бита внутреннего 57-битного аккумулятора положительной реактивной энергии

ADCUI_F2VARN

Таблица 131 Регистр ADCUI_ F2VARN

Номер	31...0
Доступ	RO
Сброс	
	F2VARHRN

Таблица 132 Описание бит регистра ADCUI_ F2VARN

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	F2VARHRN	Старшие 32 бита внутреннего 57-битного аккумулятора отрицательной реактивной энергии

ADCUI_F2AC

Таблица 133 Регистр ADCUI_ F2AC

Номер	31...28	27...16	15...12	11...0
Доступ	-	R/W	-	R/W
Сброс	-	000h	-	000h
	-	F2VAGAIN	-	F2VRMSOS

Таблица 134 Описание бит регистра ADCUI_ F2AC

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...28	-	Зарезервировано
27...16	F2VAGAIN	Калибровка усиления канала
15...12	-	Зарезервировано
11...0	F2VRMSOS	Калибровка вычислителя среднеквадратического значения напряжения

ADCUI_F2VR

Таблица 135 Регистр ADCUI_ F2VR

Номер	31...0
Доступ	R
Сброс	
	F2VAHR

Таблица 136 Описание бит регистра ADCUI_ F2VR

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	F2VAHR	Старшие 32 бита внутреннего 57-битного аккумулятора полной энергии

ADCUI_F2MD0

Таблица 137 Регистр ADCUI_ F2MD0

Номер	31..29	28..8	7	6	5	4	3..2	1..0
Доступ	-	RO	R/W	R/W	R	R	R/W	R/W
Сброс	-	-	0	0	0	0	00	00
	-	F2PER_FREQ	V2GAIN	I2GAIN	F2REACTS	F2ACTS	F2ISEL	F2VSEL

Таблица 138 Описание бит регистра ADCUI_ F2MD0

Разряды	Функциональное	Расшифровка функционального имени бита, краткое
---------	----------------	---

	имя бита	описание назначения и принимаемых значений
31...29	-	Зарезервировано
28...8	F2PER_FREQ	Длительность такта в канале напряжения
7	V2GAIN	Усиление в канале V1: 0 – нет усиления; 1 – +6 дБ усиление
6	I2GAIN	Усиление в канале I1: 0 – нет усиления; 1 – +6 дБ усиление
5	F2REACTS	Знак реактивной энергии в последний период
4	F2ACTS	Знак активной энергии в последний период
3...2	F2ISEL	Выбор источника сигнала для регистра ADCUI_F1IDAT: 00 – отсчеты тока; 01 – отсчеты активной мощности; 10 – отсчеты реактивной мощности; 11 – отсчеты полной мощности
1...0	F2VSEL	Выбор источника сигнала для регистра ADCUI_F1VDAT^ 00 – отсчеты напряжения; 01 – отсчеты активной мощности; 10 – отсчеты реактивной мощности; 11 – отсчеты полной мощности

ADCUI_F2MD1

Таблица 139 Регистр ADCUI_F2MD1

Номер	31...16	15...0
Доступ	R/W	R/W
Сброс	0000h	0000h
	F2VPKLV	F2IPKLV

Таблица 140 Описание бит регистра ADCUI_F2MD1

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...16	F2VPKLV	Предельный разрешенный уровень напряжения
15...0	F2IPKLV	Предельный разрешенный уровень тока

ADCUI_F2MD2

Таблица 141 Регистр ADCUI_F2MD2

Номер	31...17	16...0
Доступ	-	RO
Сброс	-	00000h
	-	F2PHASE

Таблица 142 Описание бит регистра ADCUI_F2MD2

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...17	-	Зарезервировано
16...0	F2PHASE	Фазовый сдвиг канала напряжения V1 по отношению к V0

ADCUI_F2VPEAK

Таблица 143 Регистр ADCUI_F2VPEAK

Номер	31...24	23...0
Доступ		R/W
Сброс		000000h
	-	F2VPEAK

Таблица 144 Описание бит регистра ADCUI_F2VPEAK

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...24	-	Зарезервировано
23...0	F2VPEAK	Пиковое значение напряжения. Обнуляется чтением из регистра.

ADCUI_F2IPEAK

Таблица 145 Регистр ADCUI_F2IPEAK

Номер	31...24	23...0
Доступ		R/W
Сброс		000000h
	-	F2IPEAK

Таблица 146 Описание бит регистра ADCUI_F2IPEAK

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...24	-	Зарезервировано
23...0	F2IPEAK	Пиковое значение тока. Обнуляется чтением из регистра.

ADCUI_F2VDAT

Таблица 147 Регистр ADCUI_F2VDAT

Номер	31...24	23...0
Доступ		R/W
Сброс		000000h
	-	F2VDAT

Таблица 148 Описание бит регистра ADCUI_F2VDAT

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...24	-	Зарезервировано
23...0	F2VDAT	FIFO отсчетов напряжения (или одной из мощностей). FIFO включается при установке любого из бит V2EN, I2EN

ADCUI_F2IDAT

Таблица 149 Регистр ADCUI_F2IDAT

Номер	31...24	23...0
Доступ		R/W

Сброс		000000h
	-	F2IDAT

Таблица 150 Описание бит регистра ADCUI_ F2IDAT

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...24	-	Зарезервировано
23...0	F2IDAT	FIFO отсчетов тока (или одной из мощностей).FIFO включается при установке любого из бит V2EN, I2EN

ADCUI_F2VRMS

Таблица 151 Регистр ADCUI_ F2VRMS

Номер	31...24	23...0
Доступ		R/W
Сброс		000000h
	-	F2VRMS

Таблица 152 Описание бит регистра ADCUI_ F2VRMS

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...24	-	Зарезервировано
23...0	F2VRMS	Среднеквадратическое значение напряжения

ADCUI_F2VRMS2

Таблица 153 Регистр ADCUI_ F2VRMS2

Номер	31...0
Доступ	RO
Сброс	
	F2VRMS2

Таблица 154 Описание бит регистра ADCUI_ F2VRMS2

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	F2VRMS2	Квадрат среднеквадратического значения напряжения

ADCUI_F2IRMS

Таблица 155 Регистр ADCUI_ F2IRMS

Номер	31...24	23...0
Доступ		R/W
Сброс		000000h
	-	F2IRMS

Таблица 156 Описание бит регистра ADCUI_ F2IRMS

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...24	-	Зарезервировано
23...0	F2IRMS	Среднеквадратическое значение тока

ADCUI_F2IRMS2

Таблица 157 Регистр ADCUI_F2IRMS2

Номер	31...0
Доступ	RO
Сброс	
	F2IRMS2

Таблица 158 Описание бит регистра ADCUI_F2IRMS2

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	F2IRMS2	Квадрат среднеквадратического значения тока

ADCUI_F2STAT

Таблица 159 Регистр ADCUI_F2STAT

Номер	6	5	4	3	2	1	0
Доступ	R/W	R/W	RO	RO	R/W	RO	RO
Сброс	0	0	0	0	0	0	0
	F2SAGF	F2IF_OVR	F2IF_FL	F2IF_EMP	F2VF_OVR	F2VF_FLL	F2VF_EMP

Номер	15	14...13	12	11	10	9	8	7
Доступ	R/W	-	R/W	R/W	R/W	R/W	R/W	R/W
Сброс	0	-	0	0	0	0	0	0
	F2APSIGN	-	F2ZXT	F2VAOV	F2VAROVP	F2WATTOVP	F2PEAKIF	F2PEAKVF

Номер	31..27	26	25	24..22	21	20	19	18	17	16
Доступ	-	R/W	R/W	-	R/W	RO	-	RO	R/W	RO
Сброс	-	-	0	-	0	0	-	0	0	0
	-	F2VAROVN	F2WATT	-	F2ZEROC	F2VANLDFL	-	F2VARNDFL	F2VARRSIGN	F2APNLD

Таблица 160 Описание бит регистра ADCUI_F2STAT

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...27	-	Зарезервировано
26	F2VAROVN	Флаг, что произошло переполнение регистра аккумулятора с отрицательной реактивной энергией. Запись "1" сбрасывает этот флаг
25	F2WATTOVN	Флаг, что произошло переполнение регистра аккумулятора

		с отрицательной активной энергией. Запись “1” сбрасывает этот флаг
24...22	-	Зарезервировано
21	F2ZEROCRS	Флаг, что произошло пересечение “0” в канале напряжения. Запись “1” сбрасывает этот флаг
20	F2VANLDFL	Полная мощность ниже уровня сравнения
19	-	Зарезервировано
18	F2VARNLDFL	Реактивная мощность ниже уровня сравнения.
17	F2VARSIGN	Смена знака реактивной мощности. Запись “1” сбрасывает этот флаг
16	F2APNLDFL	Активная мощность ниже уровня сравнения.
15	F2APSIGN	Смена знака активной мощности. Запись “1” сбрасывает этот флаг
14...13	-	Зарезервировано.
12	F2ZXTOF	Флаг, что в течении TimeOut не было пересечения напряжением значения “0”. Запись “1” сбрасывает этот флаг
11	F2VAOV	Флаг, что произошло переполнение регистра аккумулятора с полной энергией. Запись “1” сбрасывает этот флаг
10	F2VAROVP	Флаг, что произошло переполнение регистра аккумулятора с положительной реактивной энергией. Запись “1” сбрасывает этот флаг
9	F2WATTOVP	Флаг, что произошло переполнение регистра аккумулятора с положительной активной энергией. Запись “1” сбрасывает этот флаг
8	F2PEAKIF	Флаг, что произошло превышение порогового значения тока. Запись “1” сбрасывает этот флаг.
7	F2PEAKVF	Флаг, что произошло превышение порогового значения напряжения. Запись “1” сбрасывает этот флаг
6	F2SAGF	Флаг, что произошла просадка напряжения. Запись “1” сбрасывает этот флаг
5	F2IF_OVR	Флаг, что произошло переполнение FIFO F2IDAT. Запись “1” сбрасывает этот флаг
4	F2IF_FLL	Флаг, что FIFO F2IDAT заполнено
3	F2IF_EMP	Флаг, что FIFO F2IDAT пусто
2	F2VF_OVR	Флаг, что произошло переполнение FIFO F2VDAT. Запись “1” сбрасывает этот флаг
1	F2VF_FLL	Флаг, что FIFO F2VDAT заполнено
0	F2VF_EMP	Флаг, что FIFO F2VDAT пусто

ADCUI_F2MASK

Таблица 161 Регистр ADCUI_ F2MASK

Номер	6	5	4	3	2	1	0
-------	---	---	---	---	---	---	---

Доступ	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Сброс	0	0	0	0	0	0	0
	F2SAGFM	F2IF_OVRM	F2IF_FL LM	F2IF_EM PM	F2VF_O VRM	F2VF_FL LM	F2VF_E MPM

Номер	15	14...13	12	11	10	9	8	7
Доступ	R/W	-	R/W	R/W	R/W	R/W	R/W	R/W
Сброс	0	-	0	0	0	0	0	0
	F2APSI GNM	-	F2ZXTOFM	F2VAO VM	F2VAR OVPM	F2WAT TOVPM	F2PEAKIFM	F2PEAKVFM

Номер	31..27	26	25	24..22	21	20	19	18	17	16
Доступ	-	R/W	R/W	-	R/W	R/W	-	R/W	R/W	R/W
Сброс	-		0	-	0	0	-	0	0	0
	-	F2VAROVNM	F2WATTOVNM	-	F2ZEROC RSM	F2VANLDFLM	-	F2VARNLDFLM	F2VARSIGNM	F2APNLD FLM

Таблица 162 Описание бит регистра ADCUI_ F2MASK

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...27	-	Зарезервировано
26	F2VAROVNM	Маска бита F2VAROVN
25	F2WATTOVNM	Маска бита F2WATTOVN
24...22	-	Зарезервировано
21	F2ZEROC RSM	Маска бита F2ZEROCRS
20	F2VANLDFLM	Маска бита F2VANLDFL
19	-	Зарезервировано
18	F2VARNLDFLM	Маска бита F2VARNLDFL
17	F2VARSIGNM	Маска бита F2VARSIGN
16	F2APNLD FLM	Маска бита F2APNLD FLM
15	F2APSIGNM	Маска бита F2APSIGNM
14...13	-	Зарезервировано
12	F2ZXTOFM	Маска бита F2ZXTOF
11	F2VAOVM	Маска бита F2VAOV
10	F2VAROVPM	Маска бита F2VAROV P
9	F2WATTOVPM	Маска бита F2WATTOVP
8	F2PEAKIFM	Маска бита F2PEAKIF
7	F2PEAKVFM	Маска бита F2PEAKVF
6	F2SAGFM	Маска бита F2SAGF
5	F2IF_OVRM	Маска бита F2IF_OVR
4	F2IF_FLLM	Маска бита F2IF_FLL
3	F2IF_EM PM	Маска бита F2IF_EMP
2	F2VF_OVRM	Маска бита F2VF_OVR
1	F2VF_FLLM	Маска бита F2VF_FLL
0	F2VF_EM PM	Маска бита F2VF_EMP

ADCUI_CCAL1**Таблица 163 Регистр ADCUI_ CCAL1**

Номер	31...24	23...12	11...0
Доступ			
Сброс			
	-	I0BGAIN	V0BGAIN

Таблица 164 Описание бит регистра ADCUI_ CCAL1

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...24	-	Зарезервировано
23...12	I0BGAIN	Калибровочный коэффициент канала I0
11...0	V0BGAIN	Калибровочный коэффициент канала V0

ADCUI_CCAL2

Таблица 165 Регистр ADCUI_CCAL2

Номер	31...24	23...12	11...0
Доступ			
Сброс			
	-	I1BGAIN	V1BGAIN

Таблица 166 Описание бит регистра ADCUI_CCAL2

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...24	-	Зарезервировано
23...12	I1BGAIN	Калибровочный коэффициент канала I1
11...0	V1BGAIN	Калибровочный коэффициент канала V1

ADCUI_CCAL3

Таблица 167 Регистр ADCUI_CCAL3

Номер	31...24	23...12	11...0
Доступ			
Сброс			
	-	I2BGAIN	V2BGAIN

Таблица 168 Описание бит регистра ADCUI_CCAL3

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...24	-	Зарезервировано
23...12	I2BGAIN	Калибровочный коэффициент канала I2
11...0	V2BGAIN	Калибровочный коэффициент канала V2

ADCUI_CCAL4

Таблица 169 Регистр ADCUI_CCAL4

Номер	31...12	11...0
Доступ		
Сброс		
	-	I3BGAIN

Таблица 170 Описание бит регистра ADCUI_CCAL4

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...12	-	Зарезервировано
11...0	I3BGAIN	Калибровочный коэффициент канала I3

ADCUI_F0IRMS_INACTIVE

Таблица 171 Регистр ADCUI_F0IRMS_INACTIVE

Номер	31...24	23...0
-------	---------	--------

Доступ		RO
Сброс		000000h
	-	F0IRMS_INACTIVE

Таблица 172 Описание бит регистра ADCUI_ F0IRMS_INACTIVE

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...24	-	Зарезервировано
23...0	F0IRMS_INACTIVE	Среднеквадратическое значение тока неактивного канала

ADCUI_F0WATTP_L

Таблица 173 Регистр ADCUI_ F0WATTP_L

Номер	24...0
Доступ	RO
Сброс	
	F0WATTGRP_L

Таблица 174 Описание бит регистра ADCUI_ F0WATTP_L

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
24...0	F0WATTGRP	Младшие 25 бит внутреннего 57 битного аккумулятора положительной активной энергии. Необходимо производить чтение только после старшей части.

ADCUI_F0WATTN_L

Таблица 175 Регистр ADCUI_ F0WATTN_L

Номер	24...0
Доступ	RO
Сброс	
	F0WATTNRN_L

Таблица 176 Описание бит регистра ADCUI_ F0WATTN_L

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
24...0	F0WATTNRN_L	Младшие 25 бит внутреннего 57 битного аккумулятора отрицательной активной энергии. Необходимо производить чтение только после старшей части.

ADCUI_F0VARP_L

Таблица 177 Регистр ADCUI_ F0VARP_L

Номер	24..0
Доступ	RO
Сброс	
	F0VARGRP_L

Таблица 178 Описание бит регистра ADCUI_F0VARP_L

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
24...0	F0VARHRP_L	Младшие 25 бит внутреннего 57-битного аккумулятора положительной реактивной энергии. Необходимо производить чтение только после старшей части.

ADCUI_F0VARN_L

Таблица 179 Регистр ADCUI_F0VARN_L

Номер	24..0
Доступ	RO
Сброс	
	F0VARHRN_L

Таблица 180 Описание бит регистра ADCUI_F0VARN_L

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
24...0	F0VARHRN_L	Младшие 25 бит внутреннего 57-битного аккумулятора отрицательной реактивной энергии. Необходимо производить чтение только после старшей части.

ADCUI_F0VR_L

Таблица 181 Регистр ADCUI_F0VR_L

Номер	24...0
Доступ	RO
Сброс	
	F0VAHR_L

Таблица 182 Описание бит регистра ADCUI_F0VR_L

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
24...0	F0VAHR_L	Младшие 25 бит внутреннего 57-битного аккумулятора полной энергии. Необходимо производить чтение только после старшей части.

ADCUI_F1WATTP_L

Таблица 183 Регистр ADCUI_F1WATTP_L

Номер	24...0
Доступ	RO
Сброс	
	F1WATTHRP_L

Таблица 184 Описание бит регистра ADCUI_ F1WATTP_L

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
24...0	F1WATTHRP	Младшие 25 бит внутреннего 57 битного аккумулятора положительной активной энергии. Необходимо производить чтение только после старшей части.

ADCUI_F1WATTN_L

Таблица 185 Регистр ADCUI_ F1WATTN_L

Номер	24...0
Доступ	RO
Сброс	
	F1WATTHRN_L

Таблица 186 Описание бит регистра ADCUI_ F1WATTN_L

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
24...0	F1WATTHRN_L	Младшие 25 бит внутреннего 57 битного аккумулятора отрицательной активной энергии. Необходимо производить чтение только после старшей части.

ADCUI_F1VARP_L

Таблица 187 Регистр ADCUI_ F1VARP_L

Номер	24..0
Доступ	RO
Сброс	
	F1VARHRP_L

Таблица 188 Описание бит регистра ADCUI_ F1VARP_L

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
24...0	F1VARHRP_L	Младшие 25 бит внутреннего 57-битного аккумулятора положительной реактивной энергии. Необходимо производить чтение только после старшей части.

ADCUI_F1VARN_L

Таблица 189 Регистр ADCUI_ F1VARN_L

Номер	24..0
Доступ	RO
Сброс	
	F1VARHRN_L

Таблица 190 Описание бит регистра ADCUI_ F1VARN_L

Разряды	Функциональное	Расшифровка функционального имени бита, краткое
---------	----------------	---

	имя бита	описание назначения и принимаемых значений
24...0	F1VARHRN_L	Младшие 25 бит внутреннего 57-битного аккумулятора отрицательной реактивной энергии. Необходимо производить чтение только после старшей части.

ADCUI_F1VR_L

Таблица 191 Регистр ADCUI_F1VR_L

Номер	24...0
Доступ	RO
Сброс	
	F1VAHR_L

Таблица 192 Описание бит регистра ADCUI_F1VR_L

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
24...0	F1VAHR_L	Младшие 25 бит внутреннего 57-битного аккумулятора полной энергии. Необходимо производить чтение только после старшей части.

ADCUI_F2WATTP_L

Таблица 193 Регистр ADCUI_F2WATTP_L

Номер	24...0
Доступ	RO
Сброс	
	F2WATTHRP_L

Таблица 194 Описание бит регистра ADCUI_F2WATTP_L

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
24...0	F2WATTHRP	Младшие 25 бит внутреннего 57 битного аккумулятора положительной активной энергии. Необходимо производить чтение только после старшей части.

ADCUI_F2WATTN_L

Таблица 195 Регистр ADCUI_F2WATTN_L

Номер	24...0
Доступ	RO
Сброс	
	F2WATTHRN_L

Таблица 196 Описание бит регистра ADCUI_F2WATTN_L

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
24...0	F2WATTHRN_L	Младшие 25 бит внутреннего 57 битного аккумулятора отрицательной активной энергии. Необходимо производить чтение только после старшей части.

ADCUI_F2VARP_L**Таблица 197 Регистр ADCUI_F2VARP_L**

Номер	24..0
Доступ	RO
Сброс	
	F2VARHRP_L

Таблица 198 Описание бит регистра ADCUI_F2VARP_L

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
24...0	F2VARHRP_L	Младшие 25 бит внутреннего 57-битного аккумулятора положительной реактивной энергии. Необходимо производить чтение только после старшей части.

ADCUI_F2VARN_L**Таблица 199 Регистр ADCUI_F2VARN_L**

Номер	24..0
Доступ	RO
Сброс	
	F2VARHRN_L

Таблица 200 Описание бит регистра ADCUI_F2VARN_L

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
24...0	F2VARHRN_L	Младшие 25 бит внутреннего 57-битного аккумулятора отрицательной реактивной энергии. Необходимо производить чтение только после старшей части.

ADCUI_F2VR_L**Таблица 201 Регистр ADCUI_F2VR_L**

Номер	24...0
Доступ	RO
Сброс	
	F2VAHR_L

Таблица 202 Описание бит регистра ADCUI_F2VR_L

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
24...0	F2VAHR_L	Младшие 25 бит внутреннего 57-битного аккумулятора полной энергии. Необходимо производить чтение только после старшей части.

Алгоритмы вычисления окончательных результатов и их соответствия внешним сигналам

Все параметры вычисленных значений зависят от схемы включения микросхемы, а также от формата выходных данных. На рисунке ниже приведены два вида включения АЦП: полностью дифференциальное и недифференциальное включение.

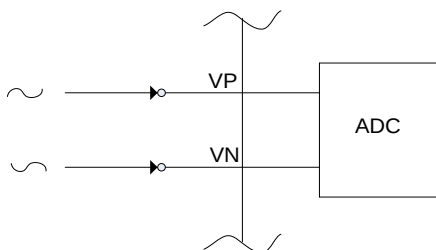


Рисунок 17. Дифференциальное включение

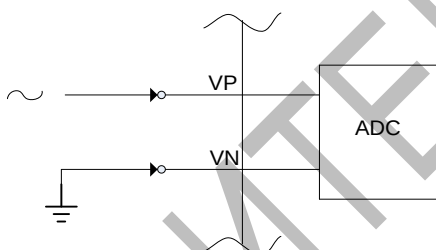


Рисунок 18. Недифференциальное включение

Необходимо иметь в виду, что значения напряжения и токов после АЦП в случае недифференциального включения в 2 раза меньше дифференциального, а мощностные характеристики в 4 раза меньше.

Для коррекции фазового сдвига в канале тока относительно канала напряжения в системе присутствует конфигурируемая линия задержки как показано ниже

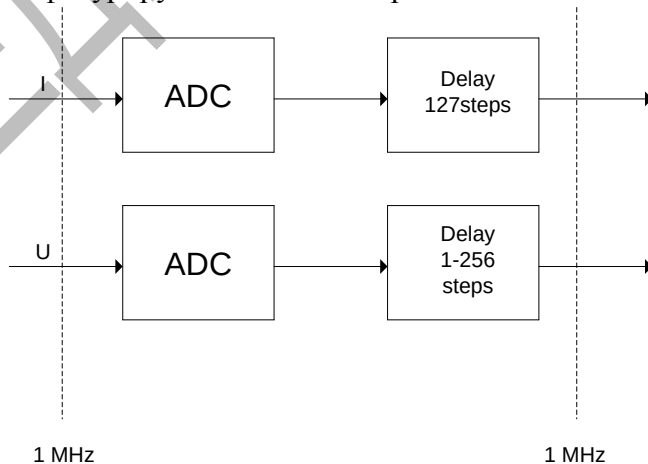


Рисунок 19. Контролируемый фазовый сдвиг в канале напряжения

Изменяя линию задержки в канале напряжения можно регулировать временной сдвиг одного канала относительно другого. Так как частота отсчетов после АЦП равна 1.024 МГц, то один шаг равен 1/20480 периода сигнала с частотой 50 Гц или 0,018 градуса. Необходимо иметь в виду, что в этой системе сдвиг осуществляется во временной области, поэтому фазовый сдвиг в градусах зависит от частоты.

В качестве децемирующего фильтра используется фильтр со структурой sinc^3 , его характеристика приведена ниже:

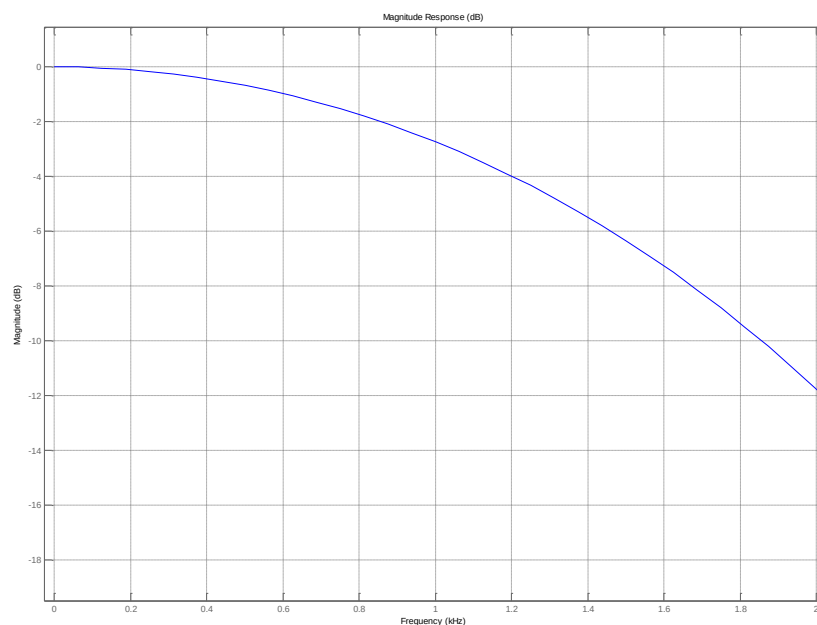


Рисунок 20. Характеристика децемирующего фильтра в полосе 2 кГц

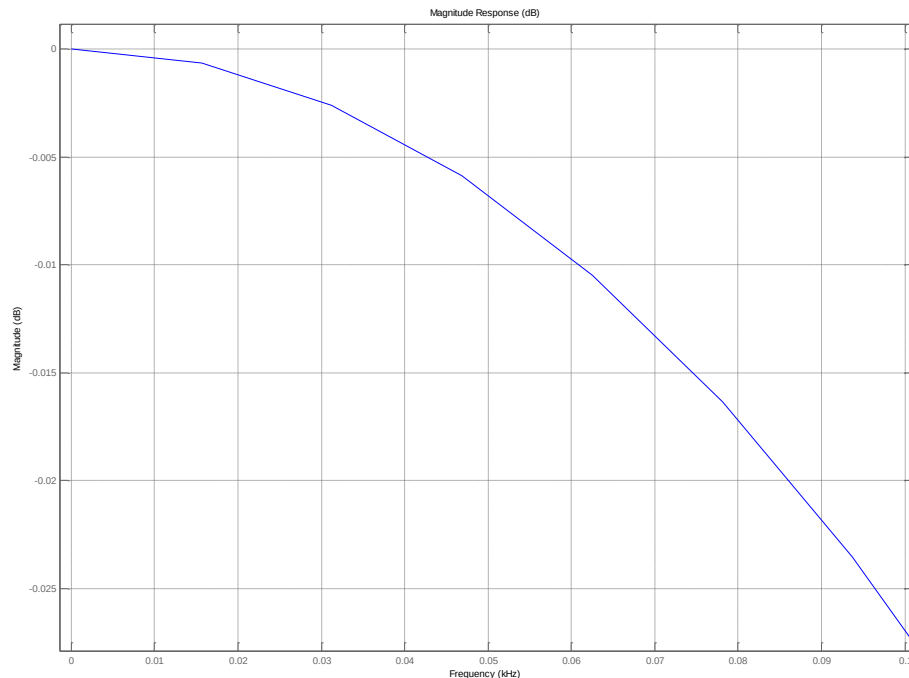


Рисунок 21. Характеристика децимирующего фильтра в полосе 100 Гц

Как видно из вышеприведенных характеристик фильтр подавляет частоты близкие к 2 кГц до величин 12 дБ, что стоит учитывать при измерении гармоник основного тона. В полосе до 100 Гц подавление незначительно (на частоте 50 Гц подавление 0,008 дБ).

Для коррекции влияния смещения в каналах тока и напряжения на вычисленную мощность после децимирующего фильтра в канале тока стоит высокочастотный фильтр. Он убирает постоянную составляющую сигнала. Частота среза фильтра равна 1 Гц.

Если FIFO каналов сконфигурированы на прием отсчетов тока и напряжение (F_xISEL=00, F_xVSEL=00), то значения отсчетов FIFO можно перевести в напряжения на входе по следующим простейшим формулам. Значения приведены для усиления PGA равному 0 дБ. Отсчеты, записанные в FIFO, представлены в двоичном формате с дополнением до 2.

Таблица 203

<i>Режим</i>	<i>Входное напряжение АЦП, Вольт</i>
Дифференциальное включение, 24 бит. режим	$\frac{F_{xV\text{DAT}}}{2^{23}}, \frac{F_{xI\text{DAT}}}{2^{23}}$
Дифференциальное включение, 16 бит. режим	$\frac{F_{xV\text{DAT}}}{2^{15}}, \frac{F_{xI\text{DAT}}}{2^{15}}$
Недифференциальное включение, 24 бит. режим	$\frac{F_{xV\text{DAT}}}{2^{22}}, \frac{F_{xI\text{DAT}}}{2^{22}}$
Недифференциальное включение, 16 бит. режим	$\frac{F_{xV\text{DAT}}}{2^{14}}, \frac{F_{xI\text{DAT}}}{2^{14}}$

Каждый из каналов тока (кроме I0) может быть скорректирован с помощью коэффициентов IBGAIN в соответствии с нижеприведенной формулой. Значение IBGAIN записывается в двоичном формате с дополнением до 2.

$$I_{COR} = I_{ADC} * (1 + \frac{IBGAIN}{2^{11}})$$

В регистрах FxVRMS и FxIRMS хранится вычисленная величина среднеквадратического значения тока и напряжения в соответствующей фазе. В таблице ниже приведены значения среднеквадратических величин.

Таблица 204

Режим	Напряжение, Вольт
Дифференциальное включение	$\frac{FxVRMS}{2^{23}}, \frac{FxIRMS}{2^{23}}$
Недифференциальное включение	$\frac{FxVRMS}{2^{22}}, \frac{FxIRMS}{2^{22}}$

Для вычисления среднеквадратического значения используется следующий алгоритм (для примера выбран канал напряжения, но для канала тока алгоритм идентичный).

Входной сигнал представлен в виде:

$$V(t) = \sqrt{2} * V_{rms} * \sin(\omega t)$$

Отсчеты напряжения поступают с частотой 4 кГц. Далее каждый отсчет возводится в квадрат, что дает следующий результат:

$$V^2(t) = 2 * V_{rms}^2 * \sin^2(\omega t) = V_{rms}^2 - V_{rms}^2 \cos(2\omega t)$$

Таким образом, мы имеем сигнал с постоянной составляющей равной среднеквадратическому значению напряжения и пульсацией с удвоенной частотой по сравнению с входным сигналом. Для фильтрации пульсации полученный сигнал пропускается через фильтр с частотой среза 2 Гц. Этот фильтр подавляет пульсации на частоте 100 Гц (50 Гц * 2) с коэффициентом 35 дБ. Отфильтрованный сигнал поступает на блок извлечения квадратного корня. Результирующий сигнал имеет так же пульсации, но ослабленные фильтром. Поэтому рекомендуется использовать режим синхронизации записи среднеквадратического значения с моментом перехода напряжения через 0 (ZXRMS=1).

После извлечения квадратного корня величину смещения среднеквадратического значения можно скорректировать с помощью 12-битных значений FxVRMSOS и FxIRMSOS. Перед корректировкой значение сдвигается на 8 бит вправо, что дает шаг корректировки в 256 меньше. Эта корректировка нужна для того, чтобы избавиться от ошибки, вызванной шумами на входе АЦП, которые после возведения в квадрат и накопления будут давать отклонения среднего уровня величины $V^2(t)$.

Формула коррекции приведена ниже:

$$V_{cor}(t) = V_{rms}(t) + \frac{FxVRMSOS}{2^{20}},$$

Значения FxVRMSOS и FxIRMSOS представлены в виде знаковых величин в двоичном коде с дополнением до 2.

В регистрах FxVRMS2 и FxIRMS2 хранятся значения среднеквадратического значения напряжения и тока до извлечения квадратного корня. В таблице ниже приведены значения квадратов среднеквадратических величин.

Таблица 205

Режим	Напряжение, Вольт^2
Дифференциальное включение	$\frac{FxVRMS^2}{2^{30}}$, $\frac{FxIRMS^2}{2^{30}}$
Недифференциальное включение	$\frac{FxVRMS^2}{2^{28}}$, $\frac{FxIRMS^2}{2^{28}}$

Для вычисления реактивной мощности необходимо сдвинуть сигнал в канале тока на 90 градусов. Это осуществляется с помощью фильтров, которые в достаточно широком диапазоне сохраняют сдвиг равный 90 градусам для обоих каналов. Ниже приведена его фазовая характеристика.

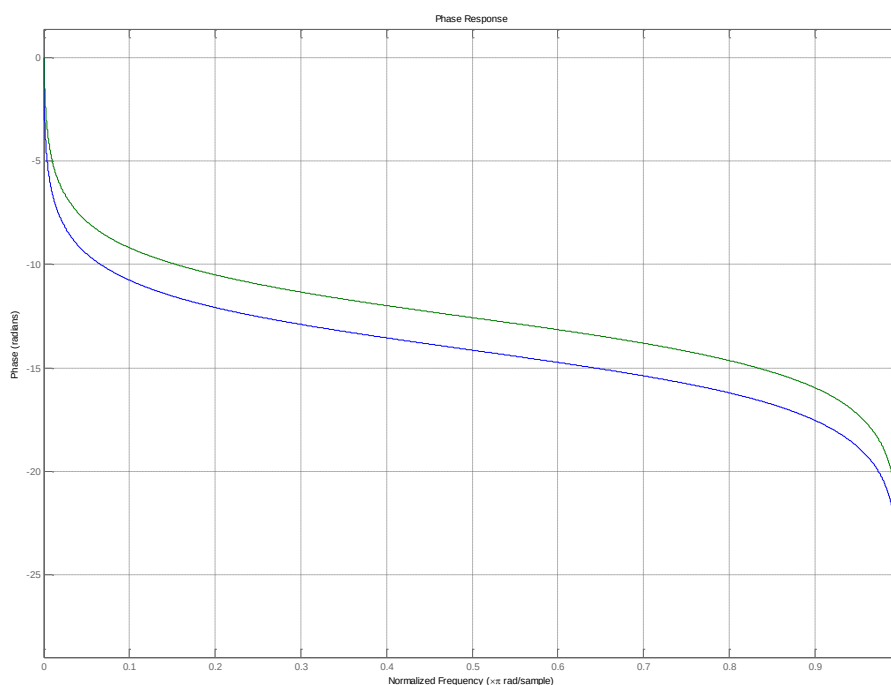


Рисунок 22. Фильтр для сдвига сигнала на 90 градусов

Для вычисления активной и реактивной энергии используется подход, похожий на вычисление среднеквадратичного значения, только без извлечения квадратного корня:

$$V(t) = \sqrt{2} * V_{rms} * \sin(\omega t)$$

$$I(t) = \sqrt{2} * I_{rms} * \sin(\omega t)$$

Тогда мгновенное значение мощности равно произведению тока на напряжение:

$$P(t) = V(t) * I(t) = V_{rms} * I_{rms} - V_{rms} * I_{rms} * \cos(2\omega t)$$

Среднее значение мощности за целое количество тактов равно:

$$P = \frac{1}{nT} \int_0^{nT} P(t) dt = V_{rms} * I_{rms}$$

Таким образом, мгновенное значение мощности равно постоянно составляющей произведения тока на напряжение. Для выделения постоянной составляющей используется низкочастотный фильтр с частотой среза 7 Гц. Значение мгновенной мощности можно получить из FIFO $FxV DAT$ и $FxI DAT$.

Если FIFO каналов сконфигурированы на прием отсчетов мощностей ($FxISEL=01/10/11$, $FxVSEL=01/10/11$), то значения отсчетов FIFO можно перевести в значения мощностей по следующим простейшим формулам. Значение приведены для усиления PGA, равного 0 дБ. Отсчеты, записанные в FIFO, представлены в двоичном формате с дополнением до 2.

Таблица 206

Режим	Мощность, Вольт*Ампер
Дифференциальное включение, 24 бит. режим	$\frac{FxV DAT}{2^{23}}$, $\frac{FxI DAT}{2^{23}}$
Дифференциальное включение, 16 бит. режим	$\frac{FxV DAT}{2^{15}}$, $\frac{FxI DAT}{2^{15}}$
Недифференциальное включение, 24 бит. режим	$\frac{FxV DAT}{2^{21}}$, $\frac{FxI DAT}{2^{21}}$
Недифференциальное включение, 16 бит. режим	$\frac{FxV DAT}{2^{13}}$, $\frac{FxI DAT}{2^{13}}$

Каждый из каналов мощности имеет независимую калибровку смещения (16 бит), а так же усиления (12 бит). Перед корректировкой смещение сдвигается на 8 бит вправо, что уменьшает шаг корректировки в 256 раз. Корректировка осуществляется в соответствии со следующей формулой:

$$P_{cor} = (P + \frac{P_{os}}{2^{23}}) * (1 + \frac{P_{gain}}{2^{11}})$$

Вычисленная мощность после калибровки накапливается в регистре аккумулятора. Для каждой из 3 мощностей есть свой аккумулятор. Значение в них определяет потребленную энергию. В таблице ниже приведена формула перевода значения в Ватт*с.

Таблица 207 Формула перевода значения в Ватт*с

Режим	Энергия, Ватт*с
Дифференциальное включение	$\frac{FxWATTHR}{512 * 4000}$, $\frac{FxWATTHR}{512 * 4000}$
Недифференциальное включение	$\frac{FxWATTHR}{512 * 1000}$, $\frac{FxWATTHR}{512 * 1000}$

Типовая схема включения для учета электроэнергии по трем фазам

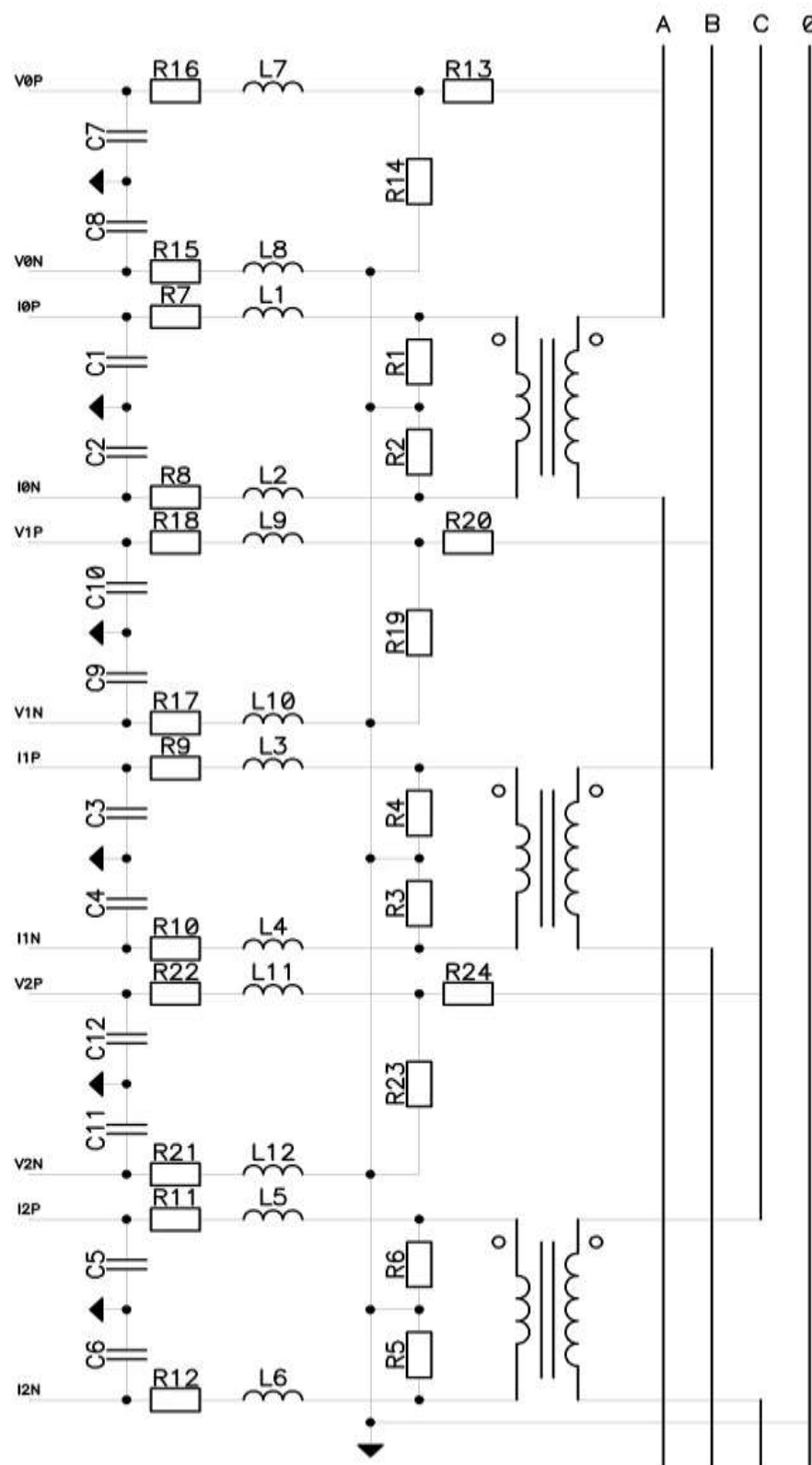


Рисунок 23. Типовая схема включения для учета электроэнергии по трем фазам

Данная схема включения может быть использована как пример для конструирования трехфазных счетчиков электроэнергии. По каждой из трех фаз установлен трансформатор тока. Выход трансформаторов нагружен на резисторный делитель. Сумма сопротивлений этих резисторов, например $R1+R2$, должна соответствовать требуемой нагрузке выхода трансформатора. Оба резистора имеют одинаковое сопротивление и создают среднюю точку, относительно которой измеряется ток. В данной схеме включения средней точкой является аналоговая земля. С этой средней точкой соединена нейтраль трехфазной сети, если она используется. Для подачи сигнала на канал напряжения используется резисторный делитель. При выборе трансформатора и расчете резисторного делителя, стоит помнить, что амплитуда сигнала на входе АЦП не должна выходить за рамки $\pm 500\text{мВ}$ относительно аналоговой земли микросхемы. Непосредственно перед входом каналов АЦП должен быть установлен антиалиасинговый фильтр, рассчитанный на частоту срезу примерно в полтора раза большую, чем частота дискретизации АЦП. В данном примере это простой RC фильтр низких частот первого порядка. Перед RC фильтром необходимо также установить индуктивности, фильтрующие радиочастотные помехи. Эти индуктивности не относятся функционально к антиалиасинговому фильтру и выбираются для наиболее широкополосного подавления радиочастот. Вы можете использовать свои варианты фильтров в зависимости от требований к конечному изделию.

Типовая схема включения для учета электроэнергии по одной фазе

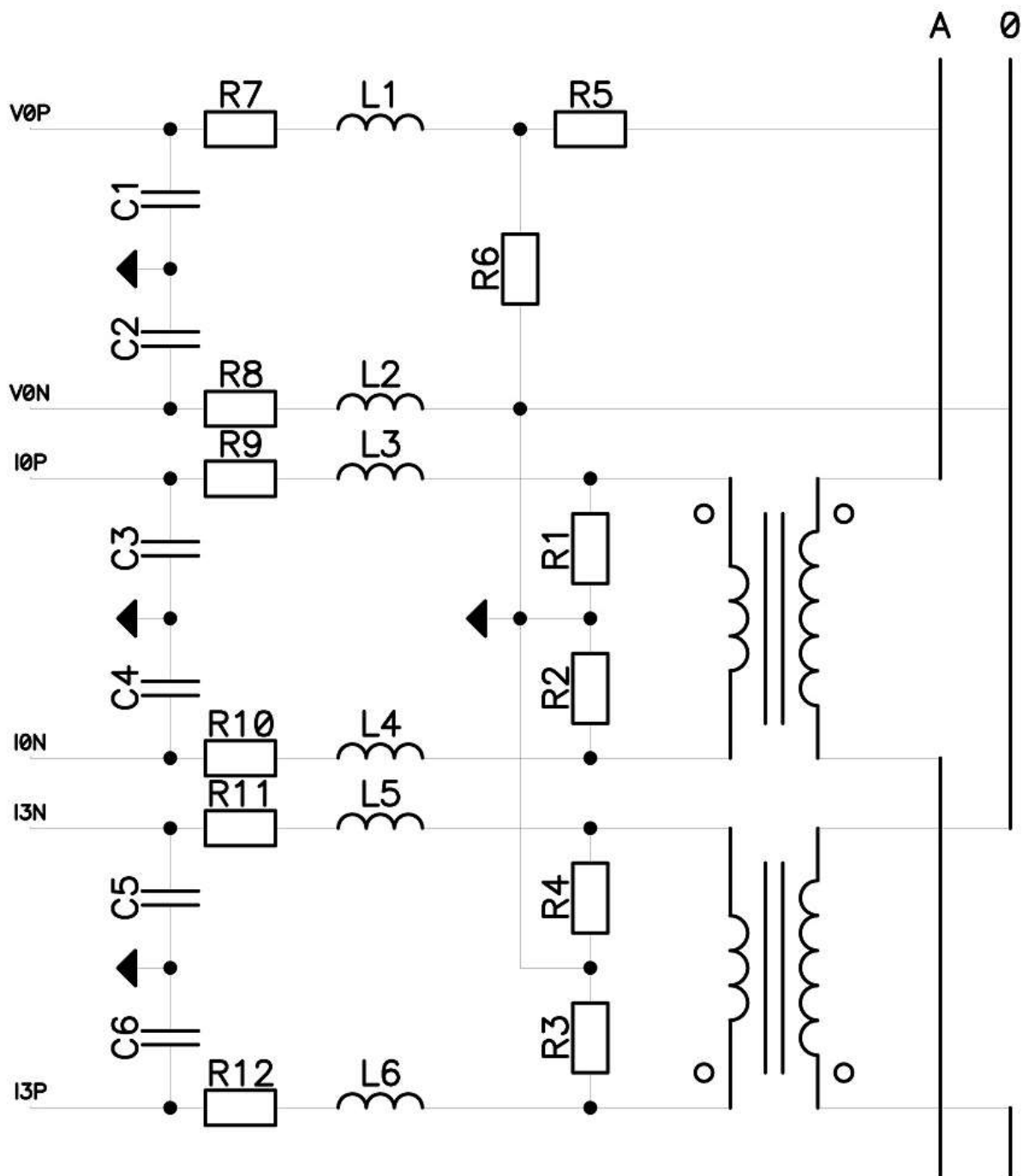


Рисунок 24. Типовая схема включения для учета электроэнергии по одной фазе

В случае, когда необходимо построить однофазный счетчик электроэнергии на базе данной микросхемы, можно взять за основу схему приведенную выше. В данной схеме предусмотрено измерение тока в обоих проводах однофазной сети. Микросхема позволяет в автоматическом режиме учитывать то значение тока из каналов I_0 и I_3 , которое будет больше. Если учета тока по «нулю» не требуется, то часть схемы, относящуюся к каналу I_3 можно убрать. В остальном, назначение элементов данной схемы аналогично схеме для учета электроэнергии по трем фазам.

Типовая схема включения для учета электроэнергии по одной фазе с использованием шунта

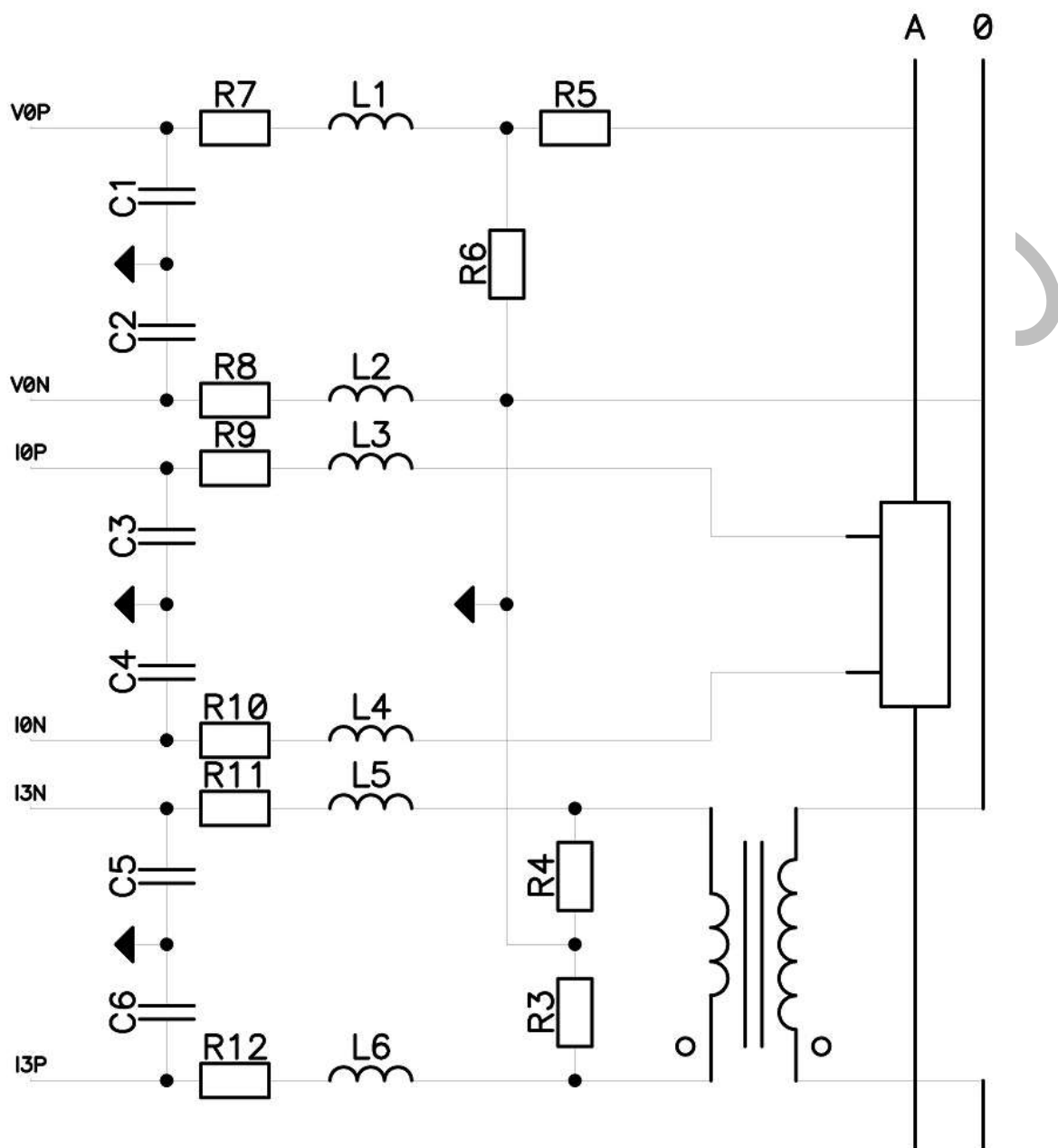


Рисунок 25. Типовая схема включения для учета электроэнергии по одной фазе с использованием шунта

Также для измерения тока может быть использован шунт. В этом случае следует использовать схему на рисунке выше. Аналогично предыдущей схеме на трансформаторах, здесь можно исключить часть схемы, относящуюся к каналу тока I3, если контроля тока в нулевом проводе не требуется.

Аппаратный блок вычисления CRC

Микроконтроллер имеет в своем составе блок для вычисления 16 битного CRC с произвольным полиномом. Контроллер принимает 32 битные слова и может их обрабатывать как в прямом порядке (начиная с младшего бита), так и в обратном (начиная со старшего

бита). Скорость подсчета составляет 2 бита / PCLK (частота APB шины). Контроллер имеет FIFO на 4 отчетов, а также DMA канал для загрузки новых слов. Запрос для DMA формируется, если в FIFO пусто. Контроллер начинает обрабатывать новые слова, как только они появляются в FIFO и обрабатывает до последнего слова. После обработки каждого слова выставляется флаг. Регистр CRC имеет доступ как на чтение (считать рассчитанное значение), так и на запись (установить начальное значение).

Описание регистров управления блока CRC

Таблица 208 Описание регистров управления блока CRC

Базовый Адрес	Название	Описание
0x4009_8000	CRC	Контроллер CRC
Смещение		
0x00	CRC_CTRL	Общее управление для контроллера CRC
0x04	CRC_STAT	Статус CRC блока
0x08	CRC_DATAI	Регистр FIFO входных данных
0x0C	CRC_VAL	Регистр подсчитанного CRC
0x10	CRC_POL	Полином для расчета CRC

CRC_CTRL

Таблица 209 Регистр CRC_CTRL

Номер	31...7	6...5	4...3	2	1	0
Доступ		R/W	R/W	R/W	R/W	R/W
Сброс		00	00	0	0	0
	-	DCSize	DLSize	DMAEN	DATAINV	CRCEN

Таблица 210 Описание бит регистра CRC_CTRL

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...7	-	Зарезервировано
6...5	DCSize	Размер данных при расчете CRC: 00 – вычисление для байта (8 бит), при этом DLSize может быть 00, 01, 10. 01 – вычисление для полуслов (16 бит), при этом DLSize может быть 01, 10. 10 – вычисление для слов (32 бит), при этом DLSize может быть только 10.
4...3	DLSize	Размер загружаемых данных: 00 – байт (8 бит), при этом загружаемый байт записывается в CRC_DATAI[7:0] 01 – полуслово (16 бит), при этом загружаемое полуслово записывается в CRC_DATAI[15:0] 10 – слово (32 бита), при этом загружаемое слово записывается в CRC_DATAI[31:0]
2	DMAEN	Разрешение формирования запроса для DMA: 0 – запрос не формируется; 1 – запрос формируется
1	DATAINV	Порядок вычисления CRC:

		0 – начиная с младшего разряда; 1 – начиная со старшего разряда
0	CRCEN	Разрешение работы блока: 0 – блок выключен; 1 – блок включен

CRC_STAT

Таблица 211 Регистр CRC_STAT

Номер	31:4	3	2	1	0
Доступ		R/W	R	R	R
Сброс		0	0	0	0
	-	FIFOOVER	FIFOEMPTY	FIFOFULL	CONVCOMP

Таблица 212 Описание бит регистра CRC_STAT

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...4	-	Зарезервировано
3	FIFOOVER*	Переполнение FIFO: 0 – корректная работа; 1 – была запись в полное FIFO, что привело к потере данных
2	FIFOEMPTY	FIFO пусто: 0 – FIFO имеет по крайней мере одну заполненную ячейку; 1 – FIFO пусто
1	FIFOFULL	FIFO заполнено: 0 – FIFO имеет по крайней мере одну свободную ячейку; 1 – FIFO не имеет свободных ячеек
0	CONVCOMP	Завершение расчета CRC: 0 – расчет идет; 1 – расчет слова данных завершен или блок отключен

* - сброс бита происходит записью “1” в разряд [3]

CRC_DATAI

Таблица 213 Регистр CRC_DATAI

Номер	31...0
Доступ	W
Сброс	
	DATA_IN

Таблица 214 Описание бит регистра CRC_DATAI

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	DATA_IN	Регистр для записи нового отчета в FIFO

CRC_VAL

Таблица 215 Регистр CRC_VAL

Номер	31...16	15...0
Доступ		R/W
Сброс		0000000000000000
	-	CRCOUT

Таблица 216 Описание бит регистра CRC_VAL

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...16	-	Зарезервировано
15...0	CRCOUT	Рассчитанное значение /начальное значение. Начальное значение нужно записывать, когда блок отключен или когда закончено преобразование

CRC_POL

Таблица 217 Регистр CRC_POL

Номер	31...17	16...0
Доступ		R/W
Сброс		1000000000000001
	-	CRCPOL

Таблица 218 Описание бит регистра CRC_POL

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...17	-	Зарезервировано
16...0	CRC_POL	Полином для расчета CRC. Так как это 16-битное CRC, то младший и старший биты всегда "1" и их нельзя изменить

Ниже приведен результирующий полином:

$$f(x) = x^{16} + x^{CRC_POL[15]} + x^{CRC_POL[14]} + \dots + x^{CRC_POL[2]} + x^{CRC_POL[1]} + 1$$

Сигналы тактовой частоты

Микроконтроллер имеет 2 встроенных генератора и 2 внешних осциллятора. А также специализированный блок формирования тактовой синхронизации микроконтроллера.

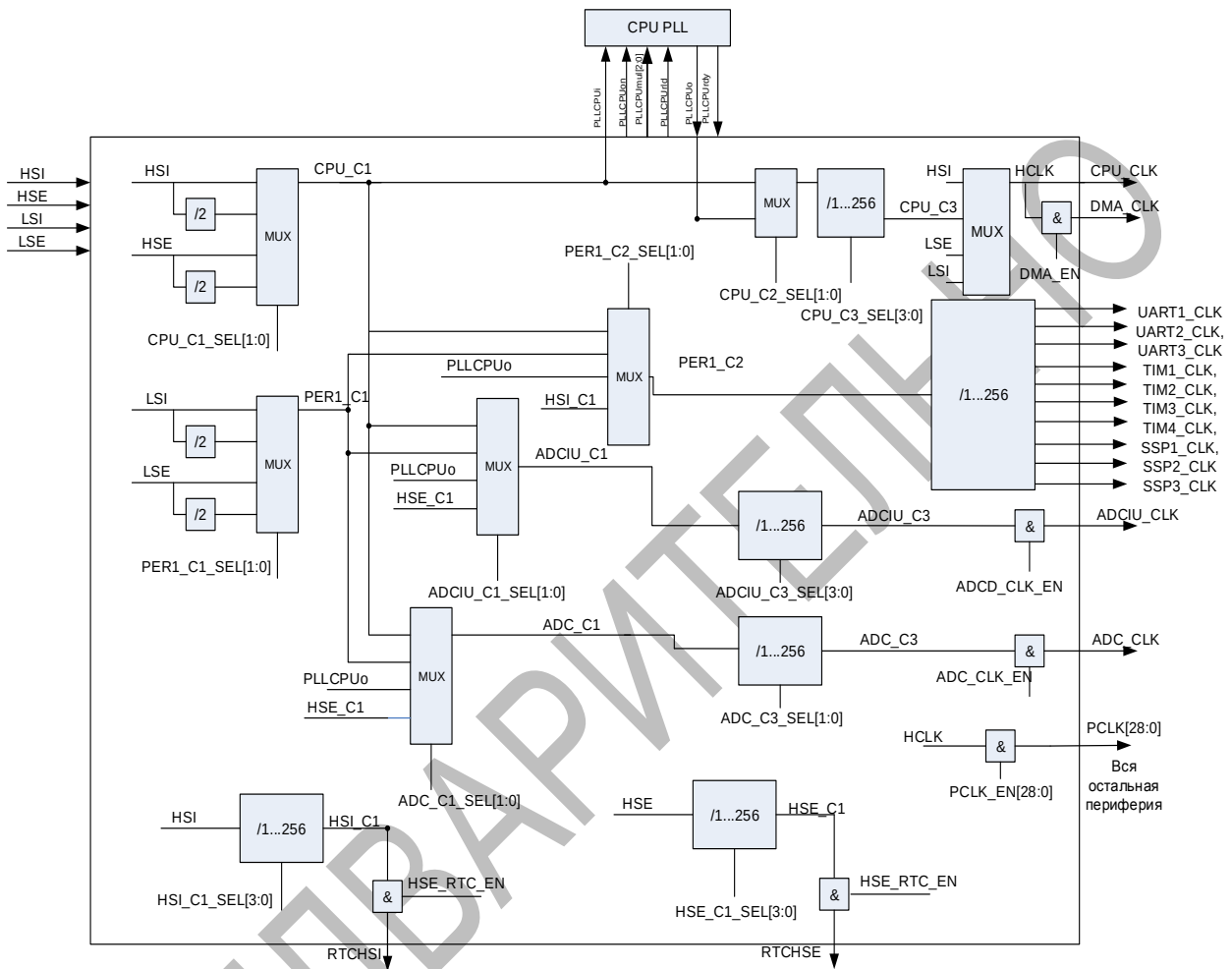


Рисунок 26. Структурная схема

Встроенный RC Генератор HSI

Генератор HSI вырабатывает тактовую частоту 8 МГц. Генератор автоматически запускается при появлении питания U_{cc} и при выходе в нормальный режим работы вырабатывает сигнал $hsirdy$ в регистре батарейного домена BKP_CLK. Первоначально процессорное ядро запускается на тактовой частоте HSI. При дальнейшей работе генератор HSI может быть отключен при помощи сигнала $hsion$ в регистре BKP_CLK. Так же генератор может быть подстроен при помощи сигнала $hsitrim$ в регистре BKP_CLK.

Встроенный RC генератор LSI

Генератор LSI вырабатывает тактовую частоту 32 кГц. Генератор включается сигналом $lsion$ в регистре BKP_CLK и при выходе в нормальный режим работы вырабатывает сигнал $lsirdy$ в регистре BKP_CLK.

Внешний осциллятор HSE

Осциллятор HSE предназначен для выработки тактовой частоты 8...16 МГц с помощью внешнего резонатора. Емкости внешних конденсаторов нагрузки 24 и 16 пФ для резонаторов на 8 и 16 МГц соответственно. Осциллятор запускается при появлении питания Ucc и сигнала разрешения HSEON в регистре HS_CONTROL. При выходе в нормальный режим работы вырабатывает сигнал HSERDY в регистре CLOCK_STATUS. Так же осциллятор может работать в режиме hsebyr, когда входная тактовая частота с входа OSC_IN проходит напрямую на выход HSE. Выход OSC_OUT находится в этом режиме в третьем состоянии. Допустимый диапазон частот в режиме hsebyr от 1 до 60 МГц.

Внешний осциллятор LSE

Осциллятор LSE предназначен для выработки тактовой частоты 32 кГц с помощью внешнего резонатора. Емкости внешних конденсаторов нагрузки от 8 до 12 пФ. Осциллятор запускается при появлении питания BDUcc и сигнала разрешения lseon в регистре BKP_CLK. При выходе в нормальный режим работы вырабатывает сигнал lserdy в регистре BKP_CLK. Так же осциллятор может работать в режиме lsebyr, когда входная тактовая частота с входа OSC_IN32 проходит напрямую на выход LSE. Допустимый диапазон частот в режиме lsebyr от 20 до 40 КГц. Выход OSC_OUT32 находится в этом режиме в третьем состоянии. Так как генератор LSE питается от напряжения питания BDUcc и его регистр управления BKP_CLK расположен в батарейном домене, то генератор может продолжать работать при пропадании основного питания Ucc. Генератор LSE используется для работы часов реального времени и календаря.

Встроенный блок умножения системной тактовой частоты

Блок умножения позволяет провести умножение входной тактовой частоты на коэффициент от 2 до 8, задаваемых на входе PLLCPUMUL[2:0] в регистре PLL_CONTROL. Входная частота блока умножителя должна быть в диапазоне 8...16 МГц выходная до 60 МГц. При выходе блока умножителя тактовой частоты в расчетный режим вырабатывается сигнал PLLCPURDY в регистре CLOCK_STATUS. Блок включается с помощью сигнала PLLCPUON в регистре PLL_CONTROL. Выходная частота может быть использована как основная частота процессора и периферии.

Управление тактовыми частотами ведется через периферийный блок RST_CLK. При включении питания микроконтроллер запускается на частоте HSI генератора. Выдача тактовых сигналов синхронизации для всех периферийных блоков кроме RST_CLK отключена. Для начала работы с нужным периферийным блоком необходимо включить его тактовую частоту в регистре PER2_CLOCK. Некоторые контроллеры интерфейсов (UART, SSP, Таймеры) могут работать на частотах отличных от частоты процессорного ядра, поэтому в соответствующих регистрах (PER1_CLOCK, UART_CLOCK, SSP_CLOCK, TIM_CLOCK) могут быть заданы их скорости работы. Для изменения тактовой частоты ядра можно перейти на другой генератор и/или воспользоваться блоком умножения тактовой частоты. Для корректной смены тактовой частоты сначала должны быть сформированы необходимые тактовые частоты и за тем осуществлено переключение на них на соответствующих мультиплексорах управляемом регистре CPU_CLOCK.

Описание регистров блока контроллера тактовой частоты

Таблица 219 Описание регистров блока контроллера тактовой частоты

Базовый Адрес	Название	Описание
0x4002_0000	RST_CLK	Контроллер тактовой частоты
Смещение		
0x00	CLOCK_STATUS	Регистр состояния блока управления тактовой частотой
0x04	PLL_CONTROL	Регистр управления блоками умножения частоты
0x08	HS_CONTROL	Регистр управления высокочастотным генератором и осциллятором
0x0C	CPU_CLOCK	Регистр управления тактовой частотой процессорного ядра
0x10	PER1_CLOCK	Регистр управления тактовой частотой периферийных блоков
0x14	ADC_CLOCK	Регистр управления тактовой частотой АЦП и $\Sigma\Delta$ АЦП
0x18	RTC_CLOCK	Регистр управления формированием высокочастотных тактовых сигналов блока RTC
0x1C	PER2_CLOCK	Регистр управления тактовой частотой периферийных блоков
0x24	TIM_CLOCK	Регистр управления тактовой частотой TIMER
0x28	UART_CLOCK	Регистр управления тактовой частотой UART
0x2C	SSP_CLOCK	Регистр управления тактовой частотой SSP

CLOCK_STATUS

Таблица 220 Регистр CLOCK_STATUS

Номер	31...3	2	1	0
Доступ	U	RO	RO	U
Сброс	0	0	0	0
	-	HSE RDY	PLL CPU RDY	

Таблица 221 Описание бит регистра CLOCK_STATUS

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
---------	-------------------------	--

31...3	-	Зарезервировано
2	HSE RDY	Флаг выхода в рабочий режим осциллятора HSE 0 – осциллятор не запущен или не стабилен 1 – осциллятор запущен и стабилен
1	PLL CPU RDY	Флаг выхода в рабочий режим CPU PLL 0 – PLL не запущена или не стабильна; 1 – PLL запущена и стабильна
0	-	Зарезервировано

PLL_CONTROL

Таблица 222 Регистр PLL_CONTROL

Номер	31...11	10...8	7...4	3	2	1	0
Доступ	U	R/W	U	R/W	R/W	U	U
Сброс	0	0000	0000	0	0	0	0
	-	PLL CPU MUL[2:0]	-	PLL CPU SEL	PLL CPU ON	-	-

Таблица 223 Описание бит регистра PLL_CONTROL

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...11	-	Зарезервировано
10...8	PLL CPU MUL[2:0]	Коэффициент умножения для CPU PLL: $PLL_{CPU0} = PLL_{CPUi} \times (PLL_{CPUMUL} + 1)$
7...4	-	Зарезервировано
3	PLL CPU SEL	Бит перезапуска выбора входной частоты PLL. 0 – частота с генератора HSE 1 – частота задаётся с помощью бит CPU_C1_SEL[1:0] регистра CPU_CLOCK.
2	PLL CPU ON	Бит включения PLL: 0 – PLL выключена; 1 – PLL включена
1	-	Зарезервировано
0	-	Зарезервировано

HS_CONTROL

Номер	31...2	1	0
Доступ	U	R/W	R/W
Сброс		0	0
	-	HSE BYP	HSE ON

Таблица 224 – Описание битов регистра HS_CONTROL

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
---------	-------------------------	--

31...2	-	Зарезервировано
1	HSE BYP	Бит управления HSE осциллятором 0 – режим осциллятора 1 – режим внешнего генератора
0	HSEON ON	Бит управления HSE осциллятором 0 – выключен 1 – включен

CPU_CLOCK

Таблица 225 Регистр CPU_CLOCK

Номер	31...10	9...8	7...4	3	2	1...0
Доступ	U	R/W	R/W	U	R/W	R/W
Сброс	0	00	0000	0	0	00
	-	HCLK SEL[1:0]	CPU C3 SEL[3:0]	-	CPU C2 SEL	CPU C1 SEL[1:0]

Таблица 226 Описание бит регистра CPU_CLOCK

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...10	-	Зарезервировано
9...8	HCLK SEL[1:0]	Биты выбора источника для HCLK: 00 – HSI; 01 – CPU_C3; 10 – LSE; 11 – LSI
7...4	CPU C3 SEL[3:0]	Биты выбора делителя для CPU_C3: 0xxx – CPU_C3 = CPU_C2; 1000 - CPU_C3 = CPU_C2 / 2; 1001 - CPU_C3 = CPU_C2 / 4; 1010 - CPU_C3 = CPU_C2 / 8; ... 1111 - CPU_C3 = CPU_C2 / 256
3	-	Зарезервировано
2	CPU C2 SEL	Биты выбора источника для CPU_C2: 0 – CPU_C1; 1 – PLLCPUo
1...0	CPU C1 SEL[1:0]	Биты выбора источника для CPU_C1: 00 – HSI; 01 – HSI/2; 10 – HSE; 11 – HSE/2

PER1_CLOCK

Таблица 227 Регистр PER1_CLOCK

Номер	31..14	13	12...10	9...8
-------	--------	----	---------	-------

Доступ	U	R/W	R/W	R/W
Сброс	0	0	000	00
	-	WDG CLK EN	WDG BRG [2:0]	WDG C2 SEL[1:0]

Номер	7...6	5	4	2...0	1...0
Доступ	R/W	R/W	R/W	R/W	R/W
Сброс	00	1	0	00	00
	WDG C1 SEL[1:0]	DMA_EN	KeyResetProg	PER1 C2 SEL	PER1 C1 SEL[1:0]

Таблица 228 Описание бит регистра PER1_CLOCK

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...14	-	Зарезервировано
13	WDG CLK EN	Разрешение тактовой частоты на WDG: 0 – нет частоты; 1 – есть частота
12...10	WDG BRG [2:0]	Делитель тактовой частоты WDG: 000 – WDG_CLK == WDG_C2; 001 – WDG_CLK == WDG_C2/2; 010 – WDG_CLK == WDG_C2/4; ... 111 – WDG_CLK == WDG_C2/128
9...8	WDG C2 SEL	Биты выбора источника для WDG_C2: 00 – CPU_C1; 01 – WDG_C1; 10 – PLLCPUo; 11 – HSI_CLK
7...6	WDG C1 SEL[1:0]	Биты выбора источника для WDG_C1: 00 – LSI; 01 – LSI/2; 10 – LSE; 11 – LSE/2
5	DMA_EN	Бит разрешения тактирования DMA контроллера
4	KeyResetProg	Бит сброса памяти криптографических ключей
3...2	PER1 C2 SEL	Биты выбора источника для PER1_C2: 00 – CPU_C1; 01 – PER1_C1; 10 – PLLCPUo; 11 – HSI_CLK
1...0	PER1 C1 SEL[1:0]	Биты выбора источника для PER1_C1: 00 – LSI; 01 – LSI/2; 10 – LSE;

		11 – LSE/2
--	--	------------

ADC_CLOCK

Таблица 229 Регистр ADC_CLOCK

Номер	31...14	13	12	11...8	7...4	3...2	1...0
Доступ	U	R/W	R/W	R/W	R/W	R/W	R/W
Сброс	0	0	0	0000	0000	00	00
	-	ADC CLK EN	ADCIU CLK EN	ADC C3 SEL[3:0]	ADCIU C3 SEL[3:0]	ADCIU C1 SEL[1:0]	ADC C1 SEL[1:0]

Таблица 230 Описание бит регистра ADC_CLOCK

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...14	-	Зарезервировано
13	ADC CLK EN	Бит разрешения выдачи тактовой частоты ADC CLK: 0 – запрещен; 1 – разрешен
12	ADCIU CLK EN	Бит разрешения выдачи тактовой частоты ADCIU CLK: 0 – запрещен; 1 – разрешен
11...8	ADC C3 SEL[3:0]	Биты выбора делителя для ADC_C3: 0xxx – ADC_C3 = ADC_C1; 1000 - ADC_C3 = ADC_C1 / 2; 1001 - ADC_C3 = ADC_C1 / 4; 1010 - ADC_C3 = ADC_C1 / 8; ... 1111 - ADC_C3 = ADC_C1 / 256
7...4	ADCIU C3 SEL[3:0]	Биты выбора делителя для ADCIU_C3: 0xxx – ADCIU_C3 = ADCIU_C1; 1000 - ADCIU_C3 = ADCIU_C1 / 2; 1001 - ADCIU_C3 = ADCIU_C1 / 4; 1010 - ADCIU_C3 = ADCIU_C1 / 8; ... 1111 - ADCIU_C3 = ADCIU_C1 / 256
3...2	ADCIU C1 SEL[1:0]	Биты выбора источника для ADCIU_C1: 00 – CPU_C1; 01 – PER1_C1; 10 – PLLCPUo; 11 – HSE_CLK
1...0	ADC C1 SEL[1:0]	Биты выбора источника для ADC_C1: 00 – CPU_C1; 01 – PER1_C1; 10 – PLLCPUo; 11 – HSE_CLK

RTC_CLOCK

Таблица 231 Регистр RTC_CLOCK

Номер	31...10	9	8	7...4	3...0
Доступ	U	R/W	R/W	R/W	R/W
Сброс	0	0	0	0000	0000
	-	HSI RTC EN	HSE RTC EN	HSI SEL[1:0]	HSE SEL[1:0]

Таблица 232 Описание бит регистра RTC_CLOCK

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...10	-	Зарезервировано
9	HSI RTC EN	Бит разрешения HSI RTC: 0 – запрещен; 1 - разрешен
8	HSE RTC EN	Бит разрешения HSE RTC: 0 – запрещен; 1 - разрешен
7...4	HSI SEL[3:0]	Биты выбора делителя для HSI_C1: 0xxx – HSI_C1 = HSI; 1000 - HSI_C1 = HSI / 2; 1001 - HSI_C1 = HSI / 4; 1010 - HSI_C1 = HSI / 8; ... 1111 - HSI_C1 = HSI / 256
3...0	HSE SEL[3:0]	Биты выбора делителя для HSE_C1: 0xxx – HSE_C1 = HSE; 1000 - HSE_C1 = HSE / 2; 1001 - HSE_C1 = HSE / 4; 1010 - HSE_C1 = HSE / 8; ... 1111 - HSE_C1 = HSE / 256

PER2_CLOCK

Таблица 233 Регистр PER2_CLOCK

Номер	30...0
Доступ	R/W
Сброс	0x00000010
	PCLK_EN[30:0]

Таблица 234 Описание бит регистра PER2_CLOCK

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
30...0	PCLK EN[29:0]	Биты разрешения тактирования периферийных блоков: 0 – запрещено; 1 – разрешено PCLK[0] – SSP1 PCLK[1] – UART1 PCLK[2] – UART2 PCLK[3] – EEPROM PCLK[4] – RST_CLK PCLK[5] – DMA PCLK[6] – I2C PCLK[7] – UART3 PCLK[8] – ADC PCLK[9] – WWDG PCLK[10] – IWDG PCLK[11] – POWER PCLK[12] – BKP PCLK[13] – ADCIU PCLK[14] – TIMER1 PCLK[15] – TIMER2 PCLK[16] – PORTA PCLK[17] – PORTB PCLK[18] – PORTC PCLK[19] – CRC PCLK[20] – зарезервировано PCLK[21] – SENSORS PCLK[22] – CLK_MEASURE PCLK[23] – RANDOM PCLK[24] – ISO7816 PCLK[25] – SSP2 PCLK[26] – SSP3 PCLK[27] – TIMER3 PCLK[28] – TIMER4 PCLK[29] – UART4 PCLK[30] – PORTD

TIM_CLOCK

Таблица 235 Регистр TIM_CLOCK

Номер	27	26	25	24	23...19	18...16	15...8	7...0
Доступ	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Сброс	0	0	0	0	00000	000	00000000	00000000
	TIM4 CLK EN	TIM3 CLK EN	TIM2 CLK EN	TIM1 CLK EN	TIM4 BRG [4:0]	TIM3 BRG [2:0]	TIM2 BRG [7:0]	TIM1 BRG [7:0]

Таблица 236 Описание бит регистра TIM_CLOCK

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...28	-	Зарезервировано
27	TIM4 CLK EN	Разрешение тактовой частоты на TIM4: 0 – нет частоты; 1 – есть частота
26	TIM3 CLK EN	Разрешение тактовой частоты на TIM3: 0 – нет частоты; 1 – есть частота
25	TIM2 CLK EN	Разрешение тактовой частоты на TIM2: 0 – нет частоты; 1 – есть частота
24	TIM1 CLK EN	Разрешение тактовой частоты на TIM1: 0 – нет частоты; 1 – есть частота
23...19	TIM4 BRG [4:0]	Делитель тактовой частоты TIM4: xx000 – TIM4_CLK == PER1_C2; xx001 – TIM4_CLK == PER1_C2/2; xx010 – TIM4_CLK == PER1_C2/4; ... xx111 – TIM4_CLK == PER1_C2/128
18...16	TIM3 BRG [2:0]	Делитель тактовой частоты TIM3: 000 – TIM3_CLK == PER1_C2; 001 – TIM3_CLK == PER1_C2/2; 010 – TIM3_CLK == PER1_C2/4; ... 111 – TIM3_CLK == PER1_C2/128
15...8	TIM2 BRG [7:0]	Делитель тактовой частоты TIM2: xxxxxx000 – TIM2_CLK == PER1_C2; xxxxxx001 – TIM2_CLK == PER1_C2/2; xxxxxx010 – TIM2_CLK == PER1_C2/4; ... xxxxxx111 – TIM2_CLK == PER1_C2/128
7...0	TIM1 BRG [7:0]	Делитель тактовой частоты TIM1: xxxxxx000 – TIM1_CLK == PER1_C2; xxxxxx001 – TIM1_CLK == PER1_C2/2; xxxxxx010 – TIM1_CLK == PER1_C2/4;

		... xxxxx111 – TIM1_CLK == PER1_C2/128
--	--	---

UART_CLOCK

Таблица 237 Регистр UART_CLOCK

Номер	30	29..27	26	25	24	23...16	15...8	7...0
Доступ	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Сброс	0	000	0	0	0	00000000	00000000	00000000
	UART4 CLK EN	UART4 BRG [2:0]	UART3 CLK EN	UART2 CLK EN	UART 1 CLK EN	UART 3 BRG [7:0]	UART 2 BRG [7:0]	UART 1 BRG [7:0]

Таблица 238 Описание бит регистра UART_CLOCK

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31	-	Зарезервировано
30	UART4 CLK EN	Разрешение тактовой частоты на UART4: 0 – нет частоты; 1 – есть частота
29...27	UART4 BRG [2:0]	Делитель тактовой частоты UART 3: 000 – UART 4_CLK == PER1_C2; 001 – UART 4_CLK == PER1_C2/2; 010 – UART 4_CLK == PER1_C2/4; ... 111 – UART 4_CLK == PER1_C2/128
26	UART3 CLK EN	Разрешение тактовой частоты на UART3: 0 – нет частоты; 1 – есть частота
25	UART2 CLK EN	Разрешение тактовой частоты на UART2: 0 – нет частоты; 1 – есть частота
24	UART1 CLK EN	Разрешение тактовой частоты на UART 1: 0 – нет частоты; 1 – есть частота
23...16	UART3 BRG [7:0]	Делитель тактовой частоты UART 3: xxxxx000 – UART 3_CLK == PER1_C2; xxxxx001 – UART 3_CLK == PER1_C2/2; xxxxx010 – UART 3_CLK == PER1_C2/4; ... xxxxx111 – UART 3_CLK == PER1_C2/128
15...8	UART2 BRG [7:0]	Делитель тактовой частоты UART 2: xxxxx000 – UART 2_CLK == PER1_C2; xxxxx001 – UART 2_CLK == PER1_C2/2; xxxxx010 – UART 2_CLK == PER1_C2/4; ... xxxxx111 – UART 2_CLK == PER1_C2/128
7...0	UART1 BRG	Делитель тактовой частоты UART: xxxxx000 – UART 1_CLK == PER1_C2;

	[7:0]	xxxxx001 – UART 1_CLK == PER1_C2/2; xxxxx010 – UART 1_CLK == PER1_C2/4; ... xxxxx111 – UART 1_CLK == PER1_C2/128
--	-------	---

SSP_CLOCK

Таблица 239 Регистр SSP_CLOCK

Номер	26	25	24	23...16	15...8	7...0
Доступ	R/W	R/W	R/W	R/W	R/W	R/W
Сброс	0	0	0	00000000	00000000	00000000
	SSP3 CLK EN	SSP2 CLK EN	SSP 1 CLK EN	SSP 3 BRG [7:0]	SSP 2 BRG [7:0]	SSP 1 BRG [7:0]

Таблица 240 Описание бит регистра SSP_CLOCK

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...27	-	Зарезервировано
26	SSP3 CLK EN	Разрешение тактовой частоты на SSP3: 0 – нет частоты; 1 – есть частота
25	SSP2 CLK EN	Разрешение тактовой частоты на SSP2: 0 – нет частоты; 1 – есть частота
24	SSP1 CLK EN	Разрешение тактовой частоты на SSP1: 0 – нет частоты; 1 – есть частота
23...16	SSP3 BRG [7:0]	Делитель тактовой частоты SSP3: xxxxx000 – SSP 3_CLK == PER1_C2; xxxxx001 – SSP 3_CLK == PER1_C2/2; xxxxx010 – SSP 3_CLK == PER1_C2/4; ... xxxxx111 – SSP 3_CLK == PER1_C2/128
15...8	SSP2 BRG [7:0]	Делитель тактовой частоты SSP2: xxxxx000 – SSP 2_CLK == PER1_C2; xxxxx001 – SSP 2_CLK == PER1_C2/2; xxxxx010 – SSP 2_CLK == PER1_C2/4; ... xxxxx111 – SSP 2_CLK == PER1_C2/128
7...0	SSP1 BRG [7:0]	Делитель тактовой частоты SSP1: xxxxx000 – SSP 1_CLK == PER1_C2; xxxxx001 – SSP 1_CLK == PER1_C2/2; xxxxx010 – SSP 1_CLK == PER1_C2/4; ... xxxxx111 – SSP 1_CLK == PER1_C2/128

Батарейный домен и часы реального времени

Блок батарейного домена предназначен для обеспечения функций календаря и часов реального времени, сохранения некоторого набора пользовательских данных при отключении основного источника питания. Так же в батарейном домене реализована функция контроля входов WAKEUP. Это позволяет, даже в отсутствии основного питания определять его состояние. При снижении питания Ucc в блоке SW происходит автоматическое переключение питания BDUcc с Ucc на BUcc. Если на BUcc имеется отдельный источник питания (батарейка), то батарейный домен остается включенным и может выполнять свои функции.

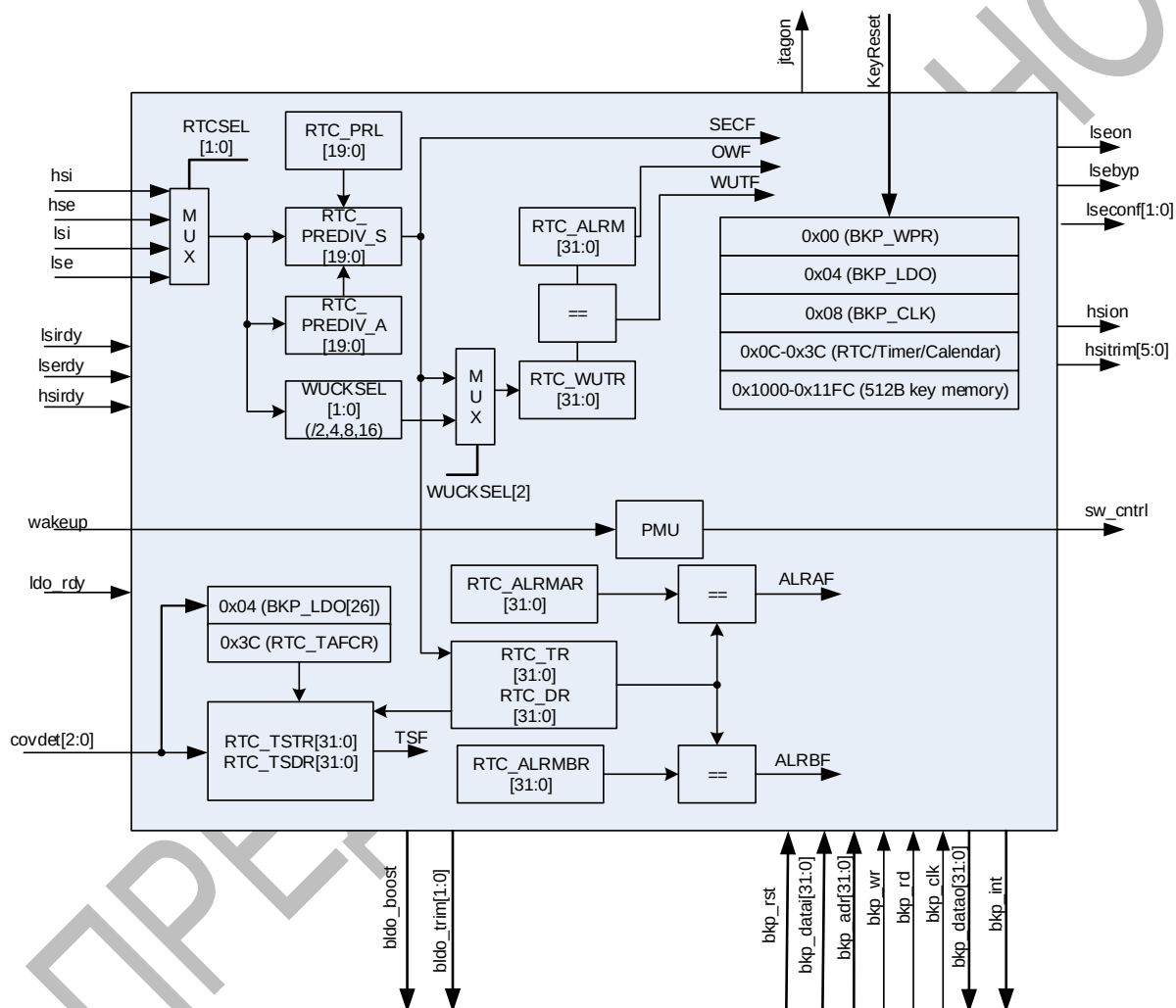


Рисунок 27. Структурная схема

Часы реального времени

Часы реального времени позволяют организовать механизм отсчета времени в кристалле, в том числе при отключении основного источника питания. Включение часов реального времени осуществляется битом RTCEN. В качестве источника тактовой частоты часов реального времени может выступать генератор LSI, осциллятор LSE, входы hse или hsi с дополнительным делителем до 256 (hse и hsi формируются в блоке управления тактовыми

частотами и могут быть выбраны только при наличии питания DUcc, LSI может быть выбран при наличии питания Ucc, LSE может быть выбран при наличии Ucc или BUcc). Выбор между источниками осуществляется битами RTCSEL. При возможном отключении основного источника питания Ucc в качестве источника тактовой частоты должен использоваться осциллятор LSE, так как он также имеет питание BDUcc. Биты управления осциллятором LSE расположены в батарейном домене и таким образом при отключении основного питания они не сбрасываются. При этом при первоначальном включении эти биты так же не определены и могут принять любое значение.

Для калибровки тактовой частоты используются биты CAL[7:0]. Значение CAL определяет, какое число тактов из 2^{20} будет замаскировано. Таким образом, с помощью бит CAL производится замедление хода часов. Для убыстрения хода часов необходимо задать меньшее, чем требуется, значение регистра RTC_PREDIV_S, а затем произвести замедление с помощью бит CAL. Изменение значения бит CAL может быть осуществлено в ходе работы часов реального времени.

Регистр RTC_PREDIV_S выступает в роли 20-ти битного предварительного делителя входной тактовой частоты таким образом, чтобы на его выходе была тактовая частота в 1 Гц. Для задания коэффициента деления регистра RTC_PREDIV_S используется регистр RTC_PRL.

Основу модуля часов реального времени составляет двоично-десятичный таймер/счетчик. Результат счета отображается в двух 32-разрядных регистрах. Первый из них RTC_TR содержит информацию о времени, второй – RTC_DR представляет собой календарь, включающий год, месяц, день недели и дату. Все данные представлены в BCD формате, что позволяет их сразу же использовать для отображения на различных индикаторах. Модуль календаря поддерживает автоматическое определение високосных лет, а также количества дней в текущем месяце.

Среди функций модуля RTC следует отметить два отдельных регистра событий (RTC_ALRMAR и RTC_ALRMBR), с помощью которых можно реализовать будильники. Формат регистров событий аналогичен счетным регистрам, что несколько упрощает их программирование. Сторожевой таймер RTC_WUTR работает независимо от часов/календаря на выходной частоте делителя RTC_PREDIV_S, либо независимой частоте, задаваемой с помощью бит WUCKSEL. Регистр RTC_ALARM предназначен для задания времени, при совпадении с которым вырабатывается флаг прерывания.

Еще одной полезной функцией может оказаться функция TimeStamp, предназначенная для определения точного времени наступления внешнего события. Для ее активации также требуется наличие внешнего сигнала, по фронту которого, значения счетных регистров сохраняются в регистрах RTC_TSTR и RTC_TSDR.

В батарейном домене реализована возможность мониторинга входных сигналов WAKEUP с помощью функции Tamper (регистр RTC_TAFCR). Во внутреннем регистре записывается контролируемый активный уровень события (передний фронт и высокий уровень или задний фронт и низкий уровень), и если сигнал на входе станет идентичным записанному, то это событие регистрируется в управляющем бите. С помощью бита sw_cntrl регистра BKP_LDO присутствует возможность разрешить выдачу сигнала управления внешним ключом на вывод SW. Внешний ключ замыкает питание батареи на основное питание микросхемы в случае его отсутствия и фиксации на любом из входов WAKEUP активного высокого или низкого уровня (в случае отключения функции Tamper, только низкого уровня на любом из входов WAKEUP). Размыкание ключа осуществляется либо при срабатывании сторожевого таймера RTC_WUTR (WUTF=1), либо при записи единицы в бит sw_off регистра BKP_LDO.

Память регистров и криптографический ключей

Батарейный домен имеет 16 встроенных 32-разрядных регистров для хранения бит управления батарейным доменом и RTC. Для разработчика программ предусмотрено ОЗУ криптографических ключей ёмкостью 512 байт.

Описание регистров блока батарейного домена

Таблица 241 Описание регистров блока батарейного домена

Базовый Адрес	Название	Описание
0x4006_0000	BKP	Контроллер батарейного домена и часов реального времени.
Смещение		
0x00	BKP_WPR	Регистр ключа для разрешения работы с регистрами блока.
0x04	BKP_LDO	Регистр управления блоком LDO
0x08	BKP_CLK	Регистр управления блоками LSE, HSE, LSI и HSI
0x0C	BKP_RTC	Регистр управления блоком RTC
0x10	RTC_WUTR	Регистр счетчика выхода из спящего режима
0x14	RTC_PREDIV_S	Регистр предварительного делителя основного счетчика/календаря
0x18	RTC_PRL	Регистр основания счета предварительного делителя основного счетчика/календаря
0x1C	RTC_ALRM	Регистр значения для сравнения счетчика выхода из спящего режима и выработки сигнала ALRF
0x20	RTC_CS	Регистр управления и состояния флагов часов реального времени
0x24	RTC_TR	Регистр времени счетчика/календаря в BCD формате
0x28	RTC_DR	Регистр даты счетчика/календаря в BCD формате
0x2C	RTC_ALRMAR	Регистр задания события А – будильника с календарным временем
0x30	RTC_ALRMBR	Регистр задания события В – будильника с календарным временем
0x34	RTC_TSTR1	Регистр времени внешнего события входа 1
0x38	RTC_TSTR2	Регистр времени внешнего события входа 2

0x3C	RTC_TSTR3	Регистр времени внешнего события входа 3
0x40	RTC_TSDR1	Регистр даты внешнего события входа 1
0x44	RTC_TSDR2	Регистр даты внешнего события входа 2
0x48	RTC_TSDR3	Регистр даты внешнего события входа 3
0x4C	RTC_TAFCR	Регистр управления обнаружением событий
0x1000-0x11FC	BKP_MEM	Память ОЗУ 512 байт

ВКР_MEM (128x32)**Таблица 242 Память ВКР_MEM**

Номер	31...0
Доступ	R/W
Сброс	X
	ВКР_MEM[31:0]

Таблица 243 Описание бит регистра ВКР_MEM

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	ВКР MEM[31:0]	Ячейка памяти ОЗУ

ВКР_WPR**Таблица 244 Память ВКР_WPR**

Номер	31...0
Доступ	R/W
Сброс	0
	ВКР_WPR[31:0]

Таблица 245 Описание бит регистра ВКР_WPR

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	ВКР WPR[31:0]	При записи в регистр значения 0x8555AAA1 открывается возможность записи в другие регистры блока батарейного домена

ВКР_LDO**Таблица 246 Регистр ВКР_LDO**

Номер	31..30	29	28..27	26	25
Доступ	U	R/W	R/W	R/W	R/W
Сброс	0	0	00	0	0
	-	sw_off	mode	covdet	fpor

Номер	15..10	9..7	6	5	4..3	2	1...0
Доступ	U	R	R/W	R/W	R/W	R/W	R/W
Сброс	0	000	1	0	00	000	000
	-	WAKEUP 3-1	jtagon	sw_cntrl	ldo_boost	blldo_boost	blldo_trim

Таблица 247 Описание бит регистра ВКР_LDO

Разряды	Функциональное	Расшифровка функционального имени бита, краткое
----------------	-----------------------	--

	имя бита	описание назначения и принимаемых значений
31...30	-	Зарезервировано
29	sw_off	Выключение аварийного питания микросхемы от батареи. Для выключения записывать единицу.
28..27	mode[1:0]	Режим запуска микроконтроллера полученный с входов MODE
26	covdet	Признак несанкционированного вскрытия устройства по входам WAKEUP3-WAKEUP1: 0-вскрытия не было; 1-осуществлялось вскрытие
25	fpor	Флаг срабатывания POR. Устанавливается в 1 загрузочным ПЗУ после сброса по питанию, при сбросе по питанию устанавливается в 0. Служит для анализа загрузочным ПЗУ, что сейчас идет выполнение программы после системного или программного сброса, либо после сброса по питанию
24..10	-	Зарезервировано
9..7	WAKEUP3-1	Состояние входов WAKEUP3-WAKEUP1
6	jtagon	Разрешение работы порта JTAG: 0 – запрещен; 1 – разрешен
5	sw_cntrl	Бит разрешения аварийного питания микросхемы от батареи при наличии активного постоянного высокого или низкого уровня на любом из входов WAKEUP1-WAKEUP3 1 - разрешено 0 - запрещено
4..3	ldo_boost	Вход активации boost режима работы основного регулятора 00 – типовое значение выхода регулятор 01...11 – увеличение выходного значения регулятора напряжения
2	blldo_boost	Вход активации boost режима работы регулятора батарейного домена 0 – типовое значение выхода регулятор 1 – увеличение выходного значения регулятора напряжения с помощью blldo_trim.
1...0	blldo_trim[1:0]	Подстройка регулятора питания батарейного домена 00 – типовое значение 01... 11 – увеличение опорного напряжения Шаг подстройки 2%

ВКР_CLK

Таблица 248 Регистр ВКР_CLK

Номер	31..21	20..15	14
Доступ	U	R/W	RO
Сброс	0	000000	1
	-	hsitrim	hsirdy

Номер	13	12	11..6	5	4..3	2	1	0
Доступ	R/W	RO	U	R/W	R/W	RO	R/W	R/W
Сброс	1	1	0	0	00	0	0	0
	hsion	lsirdy	-	lsion	lseconf	lserdy	lsebyp	lseon

Таблица 249 Описание бит регистра BKP_CLK

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31..21	-	Зарезервировано
20...15	hsitrim[5:0]	Коэффициент подстройки частоты генератора HSI. Уточняется по результатам исследования.
14	hsirdy	Флаг выхода генератора HSI в рабочий режим: 0 – генератор не запущен или не вышел в режим; 1 – генератор работает в рабочем режиме
13	hsion	Бит управления генератором HSI: 1 – генератор включен; 0 – генератор выключен
12	lsirdy	Флаг выхода генератора LSI в рабочий режим: 0 – генератор не запущен или не вышел в режим; 1 – генератор работает в рабочем режиме
11..6	-	Зарезервировано
5	lsion	Бит управления генератором LSI: 0 – генератор выключен; 1 – генератор включен
4..3	lseconf	Биты конфигурации LSE генератора. В случае возникновения проблем с запуском при дефолтном значении конфигурации lseconf[1:0]=00, следует запускать генератор при других трех конфигурациях. Остальные конфигурации менее чувствительны к помехам, но имеют больший ток потребления. Уточняется по результатам исследования.
2	lserdy	Флаг выхода генератора LSE в рабочий режим: 0 – генератор не запущен или не вышел в режим; 1 – генератор работает в рабочем режиме
1	lsebyp	Бит управления генератором LSE: 0 – режим осциллятора; 1 – режим работы на проход (внешний генератор)
0	lseon	Бит управления генератором LSE: 0 – генератор выключен; 1 – генератор включен

Номер	25	24..15	14..12	11	10..3	2	1..0
Доступ	R/W	U	R/W	R/W	R/W	R/W	R/W
Сброс	0	0	000	0	00000000	0	00
	FMT	-	WUCKSEL	RTC RESET	RTC CAL	RTC EN	RTC SEL

Таблица 251 Описание бит регистра BKP_RTC

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31..26	-	Зарезервировано
25	FMT	Формат времени: 0 -24 часа в сутках 1 -формат времени AM/PM
24..15	-	Зарезервировано
14	WUCKSEL[2]	Биты выбора независимой синхронизации для счетчика выхода из спящего режима: WUCKSEL[2] = 1 – общее тактирование со счётчиком/календарём WUCKSEL[2] = 0 – независимое тактирование с выбранного делителя
13..12	WUCKSEL[1:0]	Биты выбора делителя частоты для счетчика выхода из спящего режима: WUCKSEL[1:0] = 3 – делитель на 2 WUCKSEL[1:0] = 2 – делитель на 4 WUCKSEL[1:0] = 1 – делитель на 8 WUCKSEL[1:0] = 0 – делитель на 16
11	RTCRESET	Сброс часов реального времени: 0 – часы не сбрасываются; 1 – часы сбрасываются
10..3	RTCCAL	Коэффициент подстройки тактовой частоты часов реального времени, из каждых 2^{20} тактов будет замаскировано CAL тактов: 00000000 – 0 тактов; 00000001 – 1 такт; 11111111 – 256 тактов. Таким образом, при частоте 32768,00000 Гц: при CAL = 0 тактов, частота = 32768,00000 Гц; при CAL = 1 такт, частота = 32767,96875 Гц; ... при CAL = 256 тактов, частота = 32760,00000 Гц
2	RTCEN	Бит разрешения работы часов реального времени: 0 – работа запрещена; 1 – работа разрешена
1..0	RTCSEL[1:0]	Биты выбора источника тактовой синхронизации часов

		реального времени: 00 – lsi; 01 – lse; 10 – hsi; 11 – hse.
--	--	--

RTC_WUTR

Таблица 252 Регистр RTC_WUTR

Номер	31
Доступ	R/W
Сброс	0
	RTC WUTR[31:0]

Таблица 253 Описание бит регистра RTC_WUTR

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	RTC WUTR[31:0]	Значение основного счетчика выхода из спящего режима

RTC_PREDIV_S

Таблица 254 Регистр RTC_PREDIV_S

Номер	31...20	19...0
Доступ	U	R/W
Сброс	0	0
	-	RTC DIV [19:0]

Таблица 255 Описание бит регистра RTC_PREDIV_S

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...20	-	-
19...0	RTC DIV [19:0]	Значение счетчика предварительного делителя счетчика/календаря

RTC_PRL

Таблица 256 Регистр RTC_PRL

Номер	31...20	19...0
Доступ	U	R/W
Сброс	0	0x7FFB
	-	RTCPRL[19:0]

Таблица 257 Описание бит регистра RTC_PRL

Разряды	Функциональное	Расшифровка функционального имени бита, краткое
---------	----------------	---

	имя бита	описание назначения и принимаемых значений
31...20	-	-
19...0	RTC PRL [19:0]	Значение основания для счета счетчика предварительного делителя счетчика/календаря

RTC_ALARM

Таблица 258 Регистр RTC_ALARM

Номер	31...0
Доступ	R/W
Сброс	0
	RTC ALRM[31:0]

Таблица 259 Описание бит регистра RTC_ALARM

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	RTC ALRM[31:0]	Значения для сравнения счетчика выхода из спящего режима и выработки сигнала ALRF

RTC_CS

Таблица 260 Регистр RTC_CS

Номер	13	12	11	10	9	8	7
Доступ	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Сброс	0	0	0	0	0	0	0
	TSF	ALRB_IE	ALRA_IE	ALRBF	ALRAF	ALRBE	ALRAE

Номер	6	5	4	3	2	1	0
Доступ	RO	R/W	R/W	R/W	R/W	R/W	R/W
Сброс	0	0	0	0	0	0	0
	WEC	WUTF_IE	SECF_IE	OWF_IE	WUTF	SECF	OWF

Таблица 261 Описание бит регистра RTC_CS

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...14	-	Зарезервировано
13	TSF	Флаг внешнего события, детектированного схемой TIMESTAMP на входах WAKEUP1-WAKEUP3.
12	ALRB_IE	Разрешение прерывания по событию В: 0 - не разрешено; 1 - разрешено.
11	ALRA_IE	Разрешение прерывания по событию А: 0 - не разрешено; 1 - разрешено.
10	ALRBF	Флаг установки времени события В
9	ALRAF	Флаг установки времени события А:
8	ALRBE	Разрешение события В. Совпадения RTC_ALRMBR и

		текущего календарного времени: 0 - не разрешено; 1 - разрешено.
7	ALRAE	Разрешение события А. Совпадения RTC_ALRMAR и текущего календарного времени: 0 - не разрешено; 1 - разрешено.
6	WEC	Запись завершена: 0 – можно записывать в регистры RTC; 1 – идет запись в регистры RTC, запись в регистры запрещена.
5	WUTF_IE	Флаг разрешения прерывания по совпадению счетчика выхода из спящего режима и регистра RTC_ALRM: 0 – нет совпадения; 1 – есть совпадение
4	SECF_IE	Флаг разрешения прерывания по разрешению счета основного секундного счетчика/календаря от счетчика предварительного деления: 0 – запрет прерывания один раз в секунду; 1 – разрешение прерывания один раз в секунду
3	OWF_IE	Флаг разрешения прерывания по переполнению счетчика выхода из спящего режима RTC_WUTR: 0 – запрет прерывания по переполнению; 1 – разрешение прерывания по переполнению
2	WUTF	Флаг совпадения счетчика выхода из спящего режима и регистра RTC_ALRM: 0 – нет совпадения; 1 – есть совпадение
1	SECF	Флаг разрешения счета основного счетчика/календаря от счетчика предварительного деления: 0 – нет разрешения счета; 1 – разрешение счета
0	OWF	Флаг переполнения счетчика выхода из спящего режима RTC_WUTR: 0 – нет переполнения; 1 – было переполнение

RTC_TR

Таблица 262 Регистр RTC_TR

Номер	22	21..20	19..16	14..12	11..8	6..4	3..0
Доступ	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Сброс	0	00	0000	000	0000	000	0000
	PM	HT	HU	MNT	MNU	ST	SU

Таблица 263 Описание бит регистра RTC_TR

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
---------	-------------------------	--

31...23	-	Зарезервировано
22	PM	АМ/РМ значение: 0 – АМ или 24-часовой формат; 1 – РМ
21..20	HT[1:0]	Десятки часов в BCD формате
19..16	HU[3:0]	Единицы часов в BCD формате
15	-	Зарезервировано
14..12	MNT[2:0]	Десятки минут в BCD формате
11..8	MNU[3:0]	Единицы минут в BCD формате
7	-	Зарезервировано
6..4	ST[2:0]	Десятки секунд в BCD формате
3..0	SU[3:0]	Единицы секунд в BCD формате

RTC_DR

Таблица 264 Регистр RTC_DR

Номер	23..20	19..16	15..13	12	11..8	5..4	3..0
Доступ	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Сброс	0001	1001	010	0	0001	00	0001
	YT	YU	WDU	MT	MU	DT	DU

Таблица 265 Описание бит регистра RTC_DR

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...24	-	Зарезервировано
23..20	YT[3:0]	Десятки лет в BCD формате
19..16	YU[3:0]	Единицы лет в BCD формате
15..13	WDU[2:0]	День недели: 000 – не используется; 001 – понедельник; ... 111 – воскресенье.
12	MT	Десятки месяца в BCD формате
11..8	MU[3:0]	Единицы месяца в BCD формате
7..6	-	Зарезервировано
5..4	DT[1:0]	Десятки даты в BCD формате
3..0	DU[3:0]	Единицы даты в BCD формате

RTC_ALRMAR
RTC_ALRMBR

Таблица 266 Регистр RTC_ALRMAR
Регистр RTC_ALRMBR

Номер	31	30	29..28	27..24	23	22	21..20
Доступ	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Сброс	0	0	00	0000	0	0	00
	MSK4	WDSEL	DT	DU	MSK3	PM	HT

Номер	19..16	15	14..12	11..8	7	6..4	3..0
Доступ	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Сброс	0000	00	000	0000	0	000	0000
	HU	MSK2	MNT	MNU	MSK1	ST	SU

Таблица 267 Описание бит регистра RTC_ALRMAR
RTC_ALRMBR

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31	MSK4	Маска даты: 0 - даты учитываются; 1 - даты не учитываются.
30	WDSEL	Выбор дня недели: 0 - используется дата; 1 - DU[3:0] определяет день недели, DT[1:0] не используется)
29..28	DT[1:0]	Десятки даты в BCD формате
27..24	DU[3:0]	Единицы даты в BCD формате
23	MSK3	Маска часов: 0 - часы учитываются; 1 - часы не учитываются.
22	PM	AM/PM значение: 0 – AM или 24-часовой формат; 1 – PM
21..20	HT[1:0]	Десятки часов в BCD формате
19..16	HU[3:0]	Единицы часов в BCD формате
15	MSK2	Маска минут: 0 - минуты учитываются; 1 - минуты не учитываются.
14..12	MNT[2:0]	Десятки минут в BCD формате
11..8	MNU[3:0]	Единицы минут в BCD формате
7	MSK1	Маска секунд: 0 - секунды учитываются; 1 - секунды не учитываются.
6..4	ST[2:0]	Десятки секунд в BCD формате
3..0	SU[3:0]	Единицы секунд в BCD формате

RTC_TSTR1- RTC_TSTR3
RTC_TSDR1- RTC_TSDR3

Регистр RTC_TSTR1- RTC_TSTR3
Регистр RTC_TSDR1- RTC_TSDR3

Пара регистров, доступных только для чтения, в которых сохраняется дата наступления внешних событий. Формат полностью аналогичен соответствующим счетным регистрам RTC_TR, RTC_DR.

RTC_TAFCR

Таблица 268 Регистр RTC_TAFCR

Номер	31..22	21..19	18..16
Доступ	U	R	R
Сброс		000	000
	-	TSINSEL [2:0]	TAMP INSEL [2:0]

Номер	15..10	9	8..6	5	4	3..1	0
Доступ	U	R/W	R/W	R/W	R/W	R/W	R/W
Сброс		0	0	0	0	0	0
	-	TSIE	TSE EDGE [2:0]	TSE	TAMP IE	TAMP TRG [2:0]	TAMPE

Таблица 269 Описание бит регистра RTC_TAFCR

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...22	-	Зарезервировано
21..19	TSINSEL [2:0]	Входы, детектированные схемой TIMESTAMP : Каждый разряд отвечает за срабатывания соответствующего входа WAKEUP1 - WAKEUP3 . Если разряд установлен – вход сработал, если сброшен - вход не срабатывал. Возможна установка нескольких разрядов одновременно при срабатывании нескольких входов.
18..16	TAMPINSEL [2:0]	Входы, детектированные схемой TAMPER : Каждый разряд отвечает за срабатывания соответствующего входа WAKEUP1 - WAKEUP3 . Если разряд установлен – вход сработал, если сброшен - вход не срабатывал. Возможна установка нескольких разрядов одновременно при срабатывании нескольких входов.
15..10	-	Зарезервировано
9	TSIE	Разрешение прерывания по внешнему событию TIMESTAMP : 0 – запрещено; 1 – разрешено
8..6	TSEEDGE [2:0]	Установка фронта срабатывания для внешнего события

		TIMESTAMP: 0 - передний фронт; 1 - задний фронт
5	TSE	Разрешение сохранения времени по внешнему событию TIMESTAMP: 0 – запрещено; 1 – разрешено
4	TAMPIE	Разрешение прерывания по событию TAMPER: 0 – запрещено; 1 – разрешено
3..1	TAMPTRG[2:0]	Активный уровень события TAMPER: 0 - передний фронт и высокий уровень; 1 - задний фронт и низкий уровень
0	TAMPE	Разрешение обнаружения события TAMPER: 0 – запрещено; 1 – разрешено

Порты ввода-вывода

Микроконтроллер имеет 4 порта ввода-вывода. Порт А - 16-ти разрядный, порт В - 15-ти разрядный, порт С - 8-ми разрядный, порт D - 16-ти разрядный и их выходы мультиплексируются между различными функциональными блоками, управление для каждого вывода порта отдельное. Для того, чтобы выходы порта перешли под управление того или иного периферийного блока, необходимо задать для нужных выводов выполняемую функцию и настройки.

Таблица 270 Порты ввода-вывода

Вывод	Аналоговая функция ANALOG_EN=0	Цифровая функция					
		Порт IO	Основная		Альтернативная	Переопределённая	
		MODE=00 ANALOG_EN=1	MODE=01 ANALOG_EN=1		MODE=10 ANALOG_EN=1	MODE=11 ANALOG_EN=1	
Порт А							
PA0	-	PA0	TMR1_CH1	1	SSP2_FSS	9	-
PA1	-	PA1	TMR1_CH1N		SSP2_CLK		-
PA2	-	PA2	TMR1_CH2		SSP2_RXD		-
PA3	-	PA3	TMR1_CH2N		SSP2_TXD		-
PA4	-	PA4	TMR1_CH3		-		-
PA5	-	PA5	TMR1_CH3N		-		-
PA6	-	PA6/TCK	TMR1_CH4		-		-
PA7	-	PA7/TDO	TMR1_CH4N		-		-
PA8	-	PA8/TMS	TMR1_ETR		-		-
PA9	-	PA9/TDI	TMR1_BLK		-		-
PA10	-	PA10	EXT_INT1		UART3_RXD	10	TMR4_CH1
PA11	-	PA11	TMR2_CH4N	2	UART3_TXD		TMR4_CH1N
PA12	-	PA12	SSP1_FSS	3	-		TMR4_CH2
PA13	-	PA13	SSP1_CLK		-		TMR4_CH2N
PA14	-	PA14	SSP1_RXD		-		TMR4_CH3
PA15	-	PA15	SSP1_TXD		-		TMR4_CH3N
Порт В							
PB0	-	PB0	UART1_TXD	4	UART7816_TXD	11	TMR3_CH1
PB1	-	PB1	UART1_RXD		UART7816_RXD		TMR3_CH1N
PB2	-	PB2	nSIROUT1		UART7816_CLK		TMR3_CH2
PB3	-	PB3	nSIRIN1		UART7816_CTS		TMR3_CH2N
PB4	-	1 PB4	nUART1DTR		-		TMR3_CH3
PB5	-	PB5	nUART1RTS		UART7816_RTS	12	TMR3_CH3N
PB6	-	2 PB6	nUART1RI		EXT_INT2		TMR3_CH4
PB7	-	PB7	nUART1DCD		EXT_INT3		TMR3_CH4N
PB8	-	PB8	nUART1DSR		TMR2_ETR	13	TMR3_ETR
PB9	-	PB9	nUART1CTS		TMR2_BLK		TMR3_BLK
PB10	-	PB10	TMR2_CH2	5	UART4_RXD	20	-
PB11	-	PB11	TMR2_CH2N		UART4_TXD		-
PB12	-	PB12	TMR2_CH3		-		-
PB13	-	PB13	TMR2_CH3N		UART2_TXD	6	-
PB14	-	PB14	TMR2_CH4		UART2_RXD		-
Порт С							
PC0	-	PC0/MODE0	TMR4_ETR	19	-		TMR4_CH4
PC1	-	3 PC1/MODE1	TMR4_BLK		-		TMR4_CH4N
PC2	-	PC2	TMR2_CH1	7	SSP3_FSS	14	I2C1_SCL
PC3	-	PC3	TMR2_CH1N		SSP3_CLK		I2C1_SDA
PC4	-	PC4	EXT_INT2		SSP3_RXD		-
PC5	-	PC5	EXT_INT3		SSP3_TXD		-
PC6	-	PC6	TMR2_ETR	8	I2C1_SCL	15	-
PC7	-	PC7	TMR2_BLK		I2C1_SDA		-

Вывод	Аналоговая функция ANALOG_EN=0	Цифровая функция							
		Порт IO		Основная		Альтернативная		Переопределённая	
		MODE=00 ANALOG_EN=1		MODE=01 ANALOG_EN=1		MODE=10 ANALOG_EN=1		MODE=11 ANALOG_EN=1	
Порт D									
PD0	-		PD0	SSP2_FSS	9				
PD1	-		PD1	SSP2_CLK					
PD2	-		PD2	SSP2_RXD		-	-		
PD3	-		PD3	SSP2_TXD		-	-		
PD4	-		PD4	UART2_TXD	6	-	-		
PD5	-		PD5	UART2_RXD		-	-		
PD6	-		PD6	UART3_TXD	10	-	-		
PD7	-		PD7	UART3_RXD		-	-		
PD8	-		PD8	UART4_TXD	20	-	-		
PD9	-		PD9	UART4_RXD		-	-		
PD10	-		PD10	UART7816_TXD	11	-	-		
PD11	-		PD11	UART7816_RXD		-	-		
PD12	-		PD12	UART7816_CLK		-	-		
PD13	-		PD13	UART7816_CTS		-	-		
PD14	-		PD14	UART7816_RTS	12	-	-		
PD15	-		PD15	EXT_INT1		LSE_OUT	-		

- 1 – Таймер 1.
2, 5, 7, 8, 13 – Таймер 2.
3 – Последовательный интерфейс SSP1.
4 – Последовательный интерфейс UART1.
6 – Последовательный интерфейс UART2.
9 – Последовательный интерфейс SSP2.
10 – Последовательный интерфейс UART3.
11, 12 – Последовательный интерфейс UART ISO7816.
14 – Последовательный интерфейс SSP3.
15 – Последовательный интерфейс I2C.
16, 18, 19 – Таймер 4.
17 – Таймер 3.
20 – Последовательный интерфейс UART4.

Описание регистров портов ввода-вывода

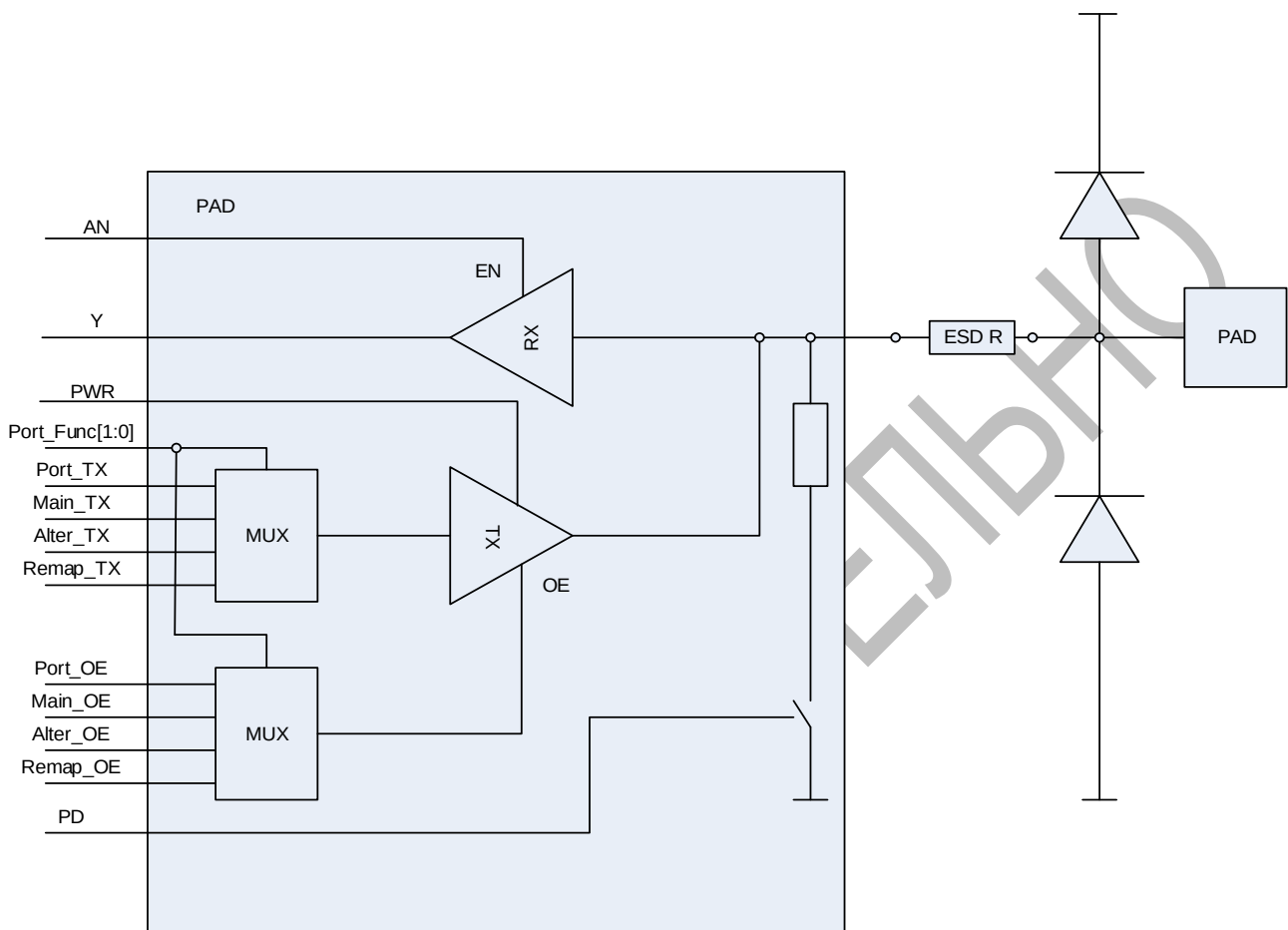


Рисунок 28. Блок схема разряда порта ввода-вывода

Таблица 271 Описание регистров портов ввода-вывода

Базовый Адрес	Название	Описание
0x4008_0000	GPIO1	Порт А
0x4008_8000	GPIO2	Порт В
0x4009_0000	GPIO3	Порт С
0x400F_0000	GPIO4	Порт D
Смещение		
0x00	PORT_RXTX[15:0]	Данные порта
0x04	PORT_OE[15:0]	Направление порта
0x08	PORT_FUNC[31:0]	Режим работы порта
0x0C	PORT_ANALOG[15:0]	Аналоговый режим работы порта
0x10	PORT_PULL[31:0]	Подтяжка порта
0x18	PORT_PWR[31:0]	Режим мощности

Базовый Адрес	Название	Описание
		передатчика
0x20	PORT_SETTX[15:0]	Регистр SET_TX записью 1 устанавливает 1 в регистре PORT_RXTX
0x24	PORT_CLRTX[15:0]	Регистр CLR_TX записью 1 устанавливает 0 в регистре RXTX
0x28	PORT_RDTX	Регистр позволяет читать то, что записано в выходной регистр порта

PORTx_RXTX

Таблица 272 Регистр PORTx_RXTX

Номер	31..16	15...0
Доступ	U	R/W
Сброс	0	0
	-	PORT RXTX[15:0]

Таблица 273 Описание бит регистра PORTx_RXTX

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...16	-	Зарезервировано
15...0	PORT RXTX[15:0]	Режим работы вывода порта Данные для выдачи на выводы порта и для чтения

PORTx_OE

Таблица 274 Регистр PORTx_OE

Номер	31...16	15...0
Доступ	U	R/W
Сброс	0	0
	-	PORT OE[15:0]

Таблица 275 Описание бит регистра PORTx_OE

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...16	-	Зарезервировано
15...0	PORT OE[15:0]	Режим работы вывода порта Направление передачи данных на выводах порта: 1 – выход; 0 - вход

PORTx_FUNC

Таблица 276 Регистр PORTx_FUNC

Номер	31	30	...	3	2	1	0
-------	----	----	-----	---	---	---	---

Доступ	R/W	R/W	...	R/W	R/W	R/W	R/W
Сброс	0	0	...	0	0	0	0
	MODE15[1:0]		...	MODE1[1:0]		MODE0[1:0]	

Таблица 277 Описание бит регистра PORTx_FUNC

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...2	MODEx	Аналогично MODE0 для остальных битов порта
1...0	MODE0[1:0]	Режим работы вывода порта: 00 – порт; 01 – основная функция; 10 – альтернативная функция 11 – переопределенная функция

PORTx_ANALOG

Таблица 278 Регистр PORTx_ANALOG

Номер	31...16	15...0
Доступ	U	R/W
Сброс	0	0
	-	ANALOG EN[15:0]

Таблица 279 Описание бит регистра PORTx_ANALOG

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...16		
15...0	ANALOG EN[15:0]	Режим работы вывода порта: 0 – аналоговый; 1 – цифровой

PORTx_PULL

Таблица 280 Регистр PORTx_PULL

Номер	31...16	15...0
Доступ	U	R/W
Сброс	-	0
	-	PULL DOWN[15:0]

Таблица 281 Описание бит регистра PORTx_PULL

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...16	-	Зарезервировано
15...0	PULL DOWN[15:0]	Режим работы вывода порта Разрешение подтяжки вниз: 1 – подтяжка в ноль включена (есть подтяжка); 0 – подтяжка в ноль выключена

PORTx_PWR

Таблица 282 Регистр PORTx_PWR

Номер	31	30	...	3	2	1	0
Доступ	U	R/W	...	U	R/W	U	R/W
Сброс	-	0	...	-	0	-	0
		PWR15			PWR1		PWR0

Таблица 283 Описание бит регистра PORTx_PWR

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...2	PWRx	Аналогично PWR0 для остальных бит порта
1	-	Зарезервировано
0	PWR0	0 – быстрый фронт (нагрузка до 2 мА) 1 – медленный фронт (нагрузка до 4 мА)

PORTx_SETTX

Таблица 284 Регистр PORTx_SETTX

Номер	31...16	15...0
Доступ	U	R/W
Сброс	0	0
	-	SETTX[15:0]

Таблица 285 Описание бит регистра PORTx_SETTX

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31..16	-	Зарезервировано
15...0	SETTX[15:0]	Регистр индивидуальной установки выхода порта Запись единицы в соответствующий разряд регистра устанавливает в единицу соответствующий разряд выхода порта PORTx_RXTX. Запись нуля не влияет на состояние соответствующего выхода порта PORTx_RXTX. Читается ранее записанным в регистр значениями, а не состояние соответствующих входов порта PORT_RXTX

PORTx_CLRTX**Таблица 286 Регистр PORTx_CLRTX**

Номер	31...16	15...0
Доступ	U	R/W
Сброс	0	0
	-	CLRTX[15:0]

Таблица 287 Описание бит регистра PORTx_CLRTX

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31..16	-	Зарезервировано
15...0	CLRTX[15:0]	Регистр индивидуального сброса выхода порта Запись единицы в соответствующий разряд регистра сбрасывает в ноль соответствующий разряд выхода порта PORTx_RXTX. Запись нуля не влияет на состояние соответствующего выхода порта PORTx_RXTX. Читается ранее записанным в регистр значениями, а не состояние соответствующих входов порта PORTx_RXTX

Детектор напряжения питания

Блок детектора напряжения питания PVD предназначен для контроля питания трёх напряжений питания при работе микроконтроллера. Блок PVD позволяет сравнивать внешние уровни напряжения с внутренними опорными уровнями и в случае превышения или снижения ниже опорного уровня выработать сигнал или прерывание для последующей программной обработки.

Уровень опорного напряжения для сравнения с напряжением основного питания задается битами PLS[2:0] в регистре PVDCS, для сравнения с батарейным питанием задается битами PLSB[2:0] в регистре PVDCS. В соответствии с уровнями напряжения формируются флаги PVD и PVDB. Данные флаги выставляются при возникновении события и сбрасываются программно.

Параметр	Не менее	Типовое	Не более
Входное основное напряжение, В	2,3	-	3,6
Входное напряжение батареи, В	1,8	-	3,6
Уровень срабатывания PVD, при PLS = "000", В		2,3	
Уровень срабатывания PVDB, при PLSB = "000", В		1,8	
Уровень срабатывания PVD, при PLS = "001", В		2,4	
Уровень срабатывания PVDB, при PLSB = "001", В		2,0	
Уровень срабатывания PVD, при PLS = "010", В		2,5	
Уровень срабатывания PVDB, при PLSB = "010", В		2,2	
Уровень срабатывания PVD, при PLS = "011", В		2,6	
Уровень срабатывания PVDB, при PLSB = "011", В		2,4	
Уровень срабатывания PVD, при PLS = "100", В		2,7	
Уровень срабатывания PVDB, при PLSB = "100", В		2,6	
Уровень срабатывания PVD, при PLS = "101", В		2,8	
Уровень срабатывания PVDB, при PLSB = "101", В		2,8	
Уровень срабатывания PVD, при PLS = "110", В		2,9	
Уровень срабатывания PVDB, при PLSB = "110", В		3,0	
Уровень срабатывания PVD при PLS = "111", В		3,0	
Уровень срабатывания PVDB,		3,2	

при PLSB = “111”, В			
---------------------	--	--	--

ПРЕДВАРИТЕЛЬНО

Описание регистров блока PVD

Таблица 288 Описание регистров блока PVD

Базовый Адрес	Название	Описание
0x4005_8000	POWER	Датчик подсистемы питания
Смещение		
0x00	PVDCS [26:0]	Регистр управления и состояния датчика питания

PVDCS

Таблица 289 Регистр PVDCS

Номер	26	25	24	23	22	21	20	19
Доступ	R/W	U	R/W	R/W	U	R/W	R/W	U
Сброс	0	0	0	0	0	0	1	0
	INV	-	INVB	IEPVD	-	IEPVDB	PVD	-

Номер	18	15..13	12..6	5..3	2	1	0
Доступ	R/W	R/W	U	R/W	R/W	U	R/W
Сброс	1	000	00000	000	0	0	0
	PVDB	PLS	-	PLSB	PVDEN	-	PVDBEN

Таблица 290 Описание бит регистра PVDCS

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...27	-	Зарезервировано
26	INV	Флаг инверсии выхода от датчика PVD: 0 – нет инверсии; 1 – есть инверсия. Если флаг не инвертируется, то он выставляется при превышении заданного уровня. Если инвертируется - то при снижении ниже заданного уровня
25	-	Зарезервировано
24	INVB	Флаг инверсии выхода от датчика PVDB: 0 – нет инверсии; 1 – есть инверсия. Если флаг не инвертируется, то он выставляется при превышении заданного уровня. Если инвертируется - то при снижении ниже заданного уровня
23	-	Всегда записывать ноль.
22	IEPVD	Флаг разрешения прерывания от датчика PVD: 0 – прерывание запрещено; 1 – прерывание разрешено.
21	IEPVDB	Флаг разрешения прерывания от датчика PVDB: 0 – прерывание запрещено; 1 – прерывание разрешено.
20	PVD	Результат сравнения напряжения основного питания: 0 – напряжение питания меньше, чем уровень, задаваемый PLS;

		1 – напряжение питания больше, чем уровень, задаваемый PLS. Очищается двойной записью 0. Если при очистке датчик продолжает выдавать сигнал, то флаг не будет очищен.
19	-	Всегда записывать ноль.
18	PVDB	Результат сравнения напряжения батарейного питания: 0 – напряжение питания меньше, чем уровень, задаваемый PLSB; 1 – напряжение питания больше, чем уровень, задаваемый PLSB Очищается двойной записью 0. Если при очистке датчик продолжает выдавать сигнал, то флаг не будет очищен.
17..16	-	Зарезервировано
15...13	PLS[2:0]	Уровень напряжения для сравнения с напряжением основного питания: 000 – более 2,3 В 001 – более 2,4 В ... 110 – более 2,9 В 111 – более 3,0 В Шаг уровня напряжения 100 мВ.
12...6	-	Зарезервировано
5...3	PLSB[2:0]	Уровень напряжения для сравнения с напряжением батарейного питания: 000 – более 1,8 В 001 – более 2,0 В ... 110 – более 3,0 В 111 – более 3,2 В Шаг уровня напряжения 200 мВ.
2	PVDEN	Бит разрешения работы блока датчика напряжения основного питания и блока OverVoltage детектора: 0 – датчик отключен; 1 – датчик включен Сбрасывается только схемой POR. Записывается только в единицу.
1	-	Зарезервировано
0	PVDBEN	Бит разрешения работы блока датчика напряжения батарейного питания: 0 – датчик отключен; 1 – датчик включен

Таймеры общего назначения

Все блоки таймеров выполнены на основе 32-битного перезагружаемого счетчика, который синхронизируется с выхода 16-битного предделителя. Перезагружаемое значение хранится в отдельном регистре. Счет может быть прямой, обратный или двунаправленный (сначала прямой до определенного значения, а затем обратный).

Каждый из четырех таймеров микроконтроллера содержит 32-битный счетчик, 16-битный предделитель частоты и 4-канальный блок захвата/сравнения. Их можно синхронизировать системной синхронизации, внешними сигналами или другими таймерами.

Помимо составляющего основу таймера счетчика, в каждый блок таймера также входит четырехканальный блок захвата/сравнения. Данный блок выполняет, как стандартные функции захвата и сравнения, так и ряд специальных функций. Таймеры имеют в своем составе 4 канала схем захвата, ШИМ с функциями формирования «мертвой зоны» и аппаратной блокировки. Каждый из таймеров может генерировать прерывания и запросы DMA.

Особенности:

- 32-битный вверх, вниз, вверх/вниз счетчик;
- 16-разрядный программируемый предварительный делитель частоты;
- до четырех независимых 32-битных каналов захвата на один таймер. Каждый из каналов захвата может захватить (скопировать) текущее значение таймера при изменении некоторого входного сигнала. В случае захвата имеется дополнительная возможность генерировать прерывание и/или запрос DMA;
- четыре 32-битных регистра сравнения (совпадения), которые позволяют осуществлять непрерывное сравнение, с дополнительной возможностью генерировать прерывание и/или запрос DMA при совпадении;
- имеется до четырёх внешних выводов, соответствующих регистрам совпадения со следующими возможностями:
 - сброс в НИЗКИЙ уровень при совпадении;
 - установка в ВЫСОКИЙ уровень при совпадении;
 - переключение (инвертирование) при совпадении;
 - при совпадении состояние выхода не изменяется;
 - переключение при некотором условии.

1.1 Функционирование

Таймер предназначен для того, чтобы подсчитывать циклы периферийной тактовой частоты F_{dts} или какие-либо внешние события и произвольно генерировать прерывания, запросы DMA или выполнять другие действия. Значения таймера, при достижении которых будут выполнены те или иные действия, задаются восьмью регистрами совпадения. Кроме того, в микроконтроллере имеются четыре входа захвата, чтобы захватить значение таймера при изменении некоторого входного сигнала, с возможностью генерировать прерывание или запрос DMA.

Структурная схема таймера представлена на рисунке 29.

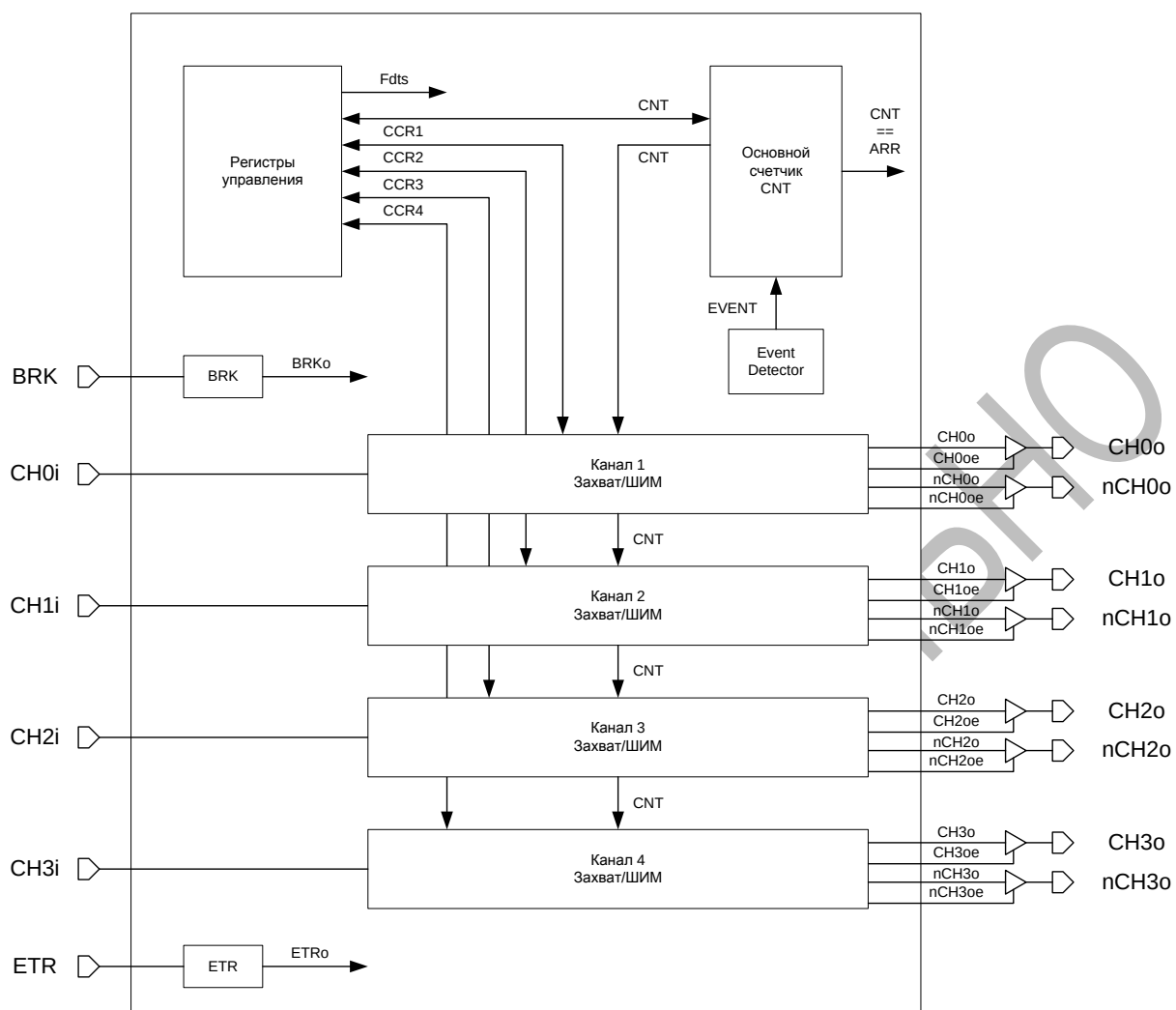


Рисунок 29 – Структурная схема таймера

Таймер содержит основной 32-битный счетчик CNT, блок регистров управления и четыре канала схем Захвата/ШИМ.

Таймер позволяет работать в режимах:

- таймер;
- расширенный таймер, с объединением нескольких таймеров;
- схема захвата;
- схема ШИМ.

Инициализация таймера

Перед началом работы с таймерами в первую очередь должны быть включены тактовые сигналы. Параметры задаются в блоке «Сигналы сброса и тактовой частоты».

Для задания тактовой частоты блока необходимо установить бит разрешения тактирования блока (бит 14 для таймера 1, бит 15 для таймера 2, бит 16 для таймера 3, бит 19 для таймера 4 регистра PER_CLOCK). В регистре TIM_CLOCK установить бит TIMyCLKEN

(для таймера 4 в регистре UART_CLOCK), чтобы разрешить тактирование определенного таймера, задать коэффициент деления тактовой частоты HCLK для каждого таймера.

После подачи тактового сигнала на блок таймера можно приступить к работе с ним.

Режим таймера

Таймеры построены на базе 32-битного счетчика, объединенного с 16-битным предварительным делителем. Скорость счета таймера зависит от значения, находящегося в регистре делителя.

Счетчик может считать вверх, вниз или сначала вверх и затем вниз.

Базовый блок таймера включает в себя:

- основной счетчик таймера (CNT);
- делитель частоты при счете основного счетчика (PSG);
- основание счета основного счетчика (ARR).

Сигналом для изменения CNT может служить как внутренняя частота TIM_CLK, так и события в других счетчиках, либо события на линиях TxCH0 данного счетчика.

Чтобы запустить работу основного счетчика необходимо задать:

- начальное значение основного счетчика таймера – CNT;
- значение предварительного делителя счетчика – PSG, при этом основной счетчик будет считать на частоте $CLK = CLK / (PSG + 1)$;
- значение основания счета для основного счетчика ARR;
- режим работы счетчика CNTRL:
 - выбрать источник события переключения счетчика EVENT_SEL;
 - режим счета основного счетчика CNT_MODE (значения 00 и 01 при тактировании внутренней частотой, значения 10 и 11 при тактировании внешними сигналами);
 - направление счета основного счетчика DIR;
- разрешить работу счетчика CNT_EN.

По событиям совпадения значения основного счетчика со значением нуля или значением основания счета генерируется прерывание и запрос DMA, которые могут быть замаскированы.

1.2 Режимы счета

Счет вверх: CNT_MODE = 00, DIR = 0.

```
TIMERx->CNTRL = 0x00000000; //Режим инициализации таймера
//Настраиваем работу основного счетчика
TIMERx->CNT = 0x00000004; //Начальное значение счетчика
TIMERx->PSG = 0x00000000; //Предделитель частоты
TIMERx->ARR = 0x00000013; //Основание счета

//Разрешение работы таймера.
TIMERx->CNTRL = 0x00000001; //Счет вверх по TIM_CLK.
```

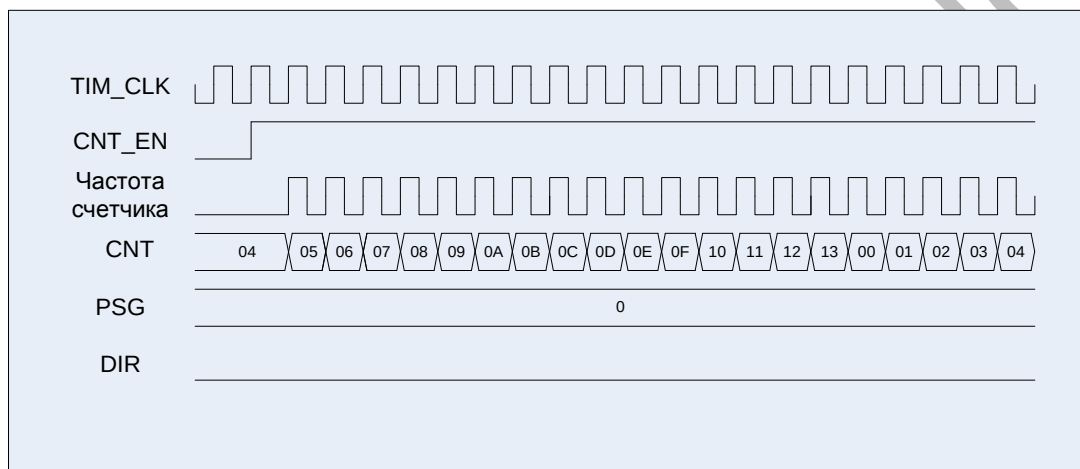


Рисунок 30 – Диаграммы работы таймера, счет вверх от 0 до 0x0013, стартовое значение 0x0004

Счет вниз: CNT_MODE = 00, DIR = 1

```
TIMERx->CNTRL = 0x00000000; //Режим инициализации таймера
//Настраиваем работу основного счетчика
TIMERx->CNT = 0x00000004; //Начальное значение счетчика
TIMERx->PSG = 0x00000000; //Предделитель частоты
TIMERx->ARR = 0x00000013; //Основание счета
```

```
//Разрешение работы таймера.
TIMERx->CNTRL = 0x00000009; //Счет вниз по TIM_CLK.
```

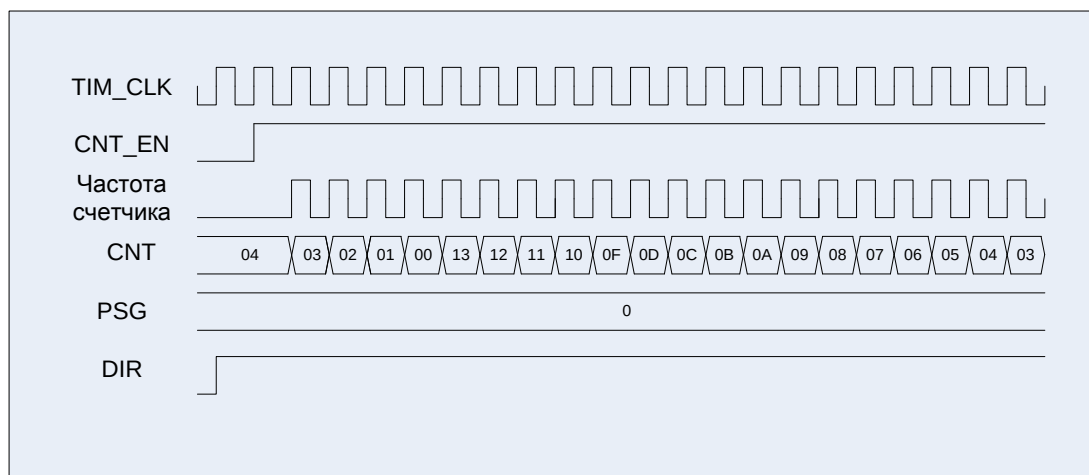


Рисунок 31 – Диаграммы работы таймера, счет вниз от 0x0013 до 0, стартовое значение 0x0004

Счет вверх/вниз: CNT_MODE = 01, DIR = 0

```
TIMERx->CNTRL = 0x00000000; //Режим инициализации таймера
//Настраиваем работу основного счетчика
TIMERx->CNT = 0x00000004; //Начальное значение счетчика
TIMERx->PSG = 0x00000000; //Предделитель частоты
TIMERx->ARR = 0x00000013; //Основание счета
```

```
//Разрешение работы таймера.
TIMERx->CNTRL = 0x00000041; //Счет вверх/вниз по TIM_CLK.
```

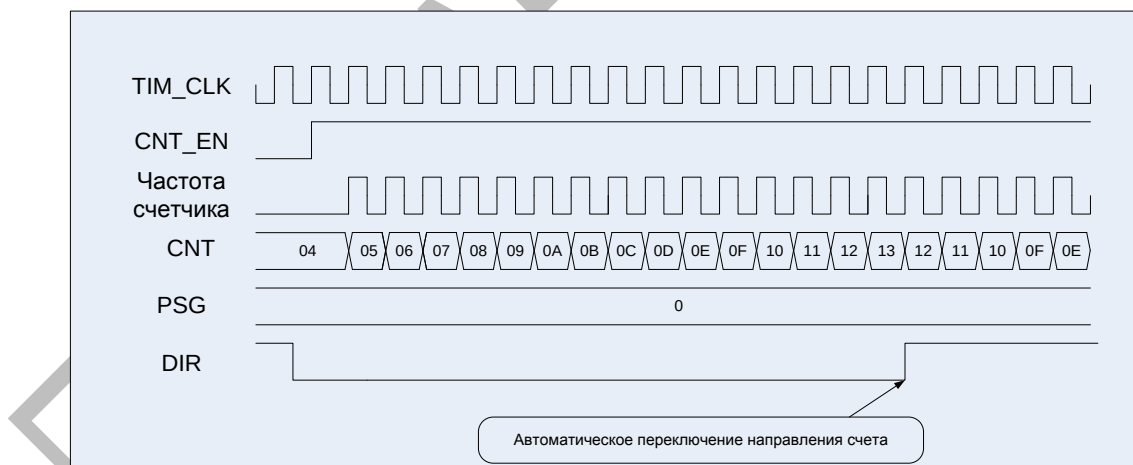


Рисунок 32 – Диаграммы работы таймера, счет вверх/вниз, сначала вверх

Счет вверх/вниз: CNT_MODE = 01, DIR = 1

```
TIMERx->CNTRL = 0x00000000; //Режим инициализации таймера
//Настраиваем работу основного счетчика
TIMERx->CNT = 0x00000004; //Начальное значение счетчика
TIMERx->PSG = 0x00000000; //Предделитель частоты
TIMERx->ARR = 0x00000013; //Основание счета
```

//Разрешение работы таймера.

`TIMEx->CNTRL = 0x00000049;`//Счет вверх/вниз по `TIM_CLK`.

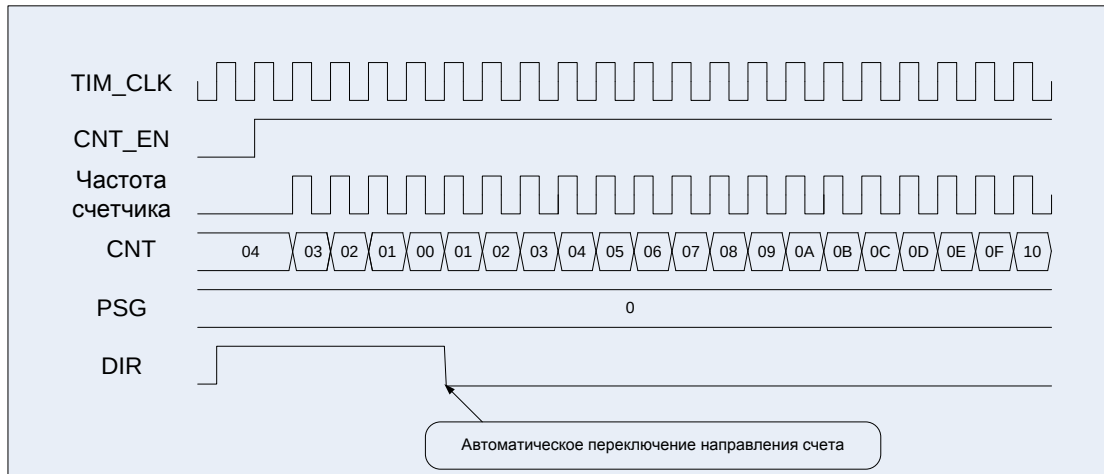


Рисунок 33 – Диаграммы работы таймера, счет вверх/вниз, сначала вниз

1.3 Источник событий для счета

Источники событий для счета:

- внутренний тактовый сигнал (`TIM_CLK`);
- события в других счетчиках (`CNT==ARR` в таймере X);
- внешний тактовый сигнал режим 1: События на линиях `TxCHO` данного счетчика;
- внешний тактовый сигнал режим 2: События на линиях `TxCHO` данного счетчика;
- внешний тактовый сигнал режим 3: События на входе `ETR` данного счетчика.

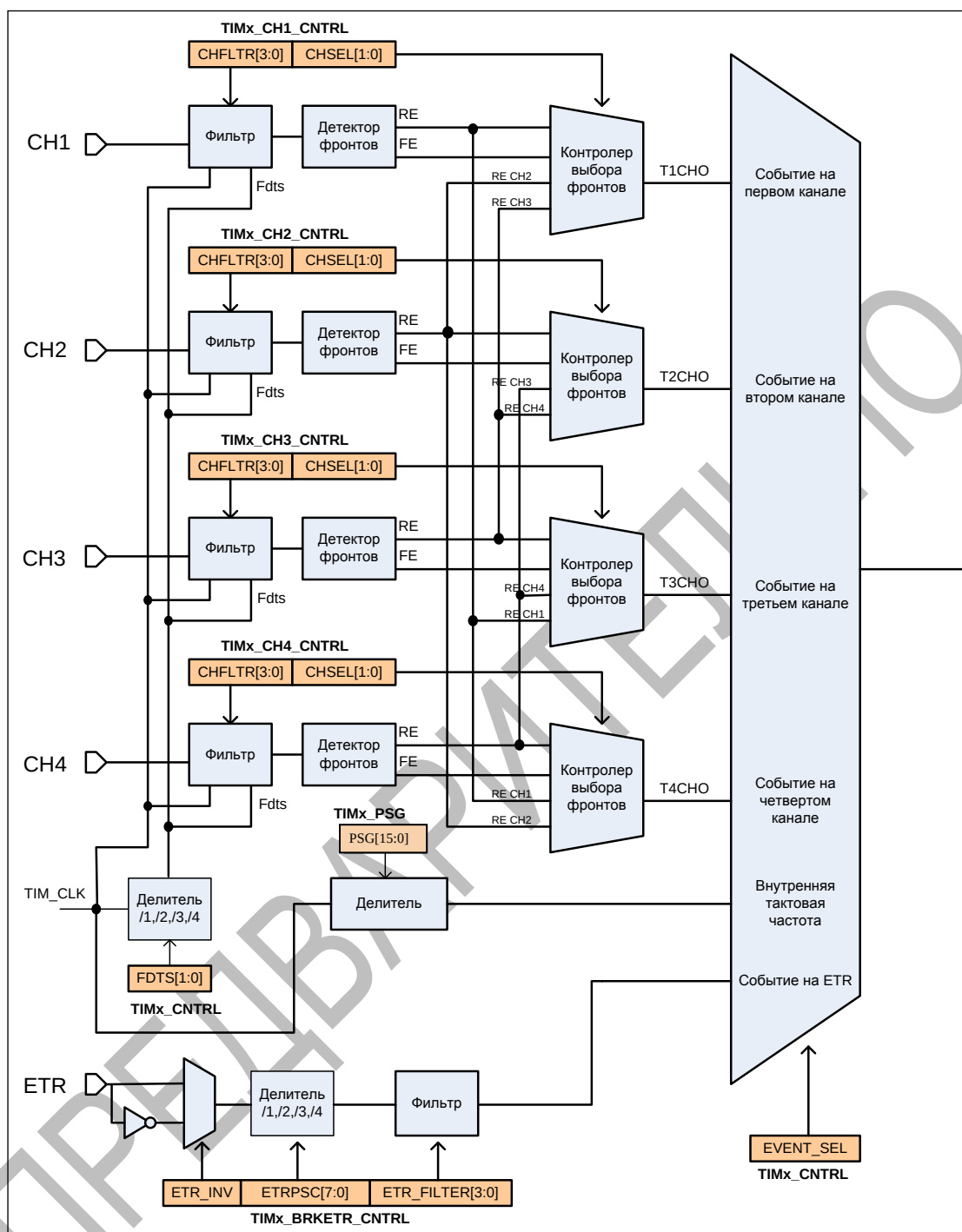


Рисунок 34 – Структурная схема формирования события для счета

Внутренний тактовый сигнал (TIM_CLK)

Этот режим выбирается, когда $CNT_MODE = 0x$, $EVENT_SEL = 0000$. Для запуска этого режима необходимо задать начальное значение основного счетчика, значение предварительного делителя основного счетчика, основание счета для основного счетчика и задать режим работы в регистре CNTRL. Значения регистров CNT, PSG и ARR можно

изменять даже во время работы счетчика, при этом их значения вступают в силу по $CNT = ARR$ или $CNT = 0$, в зависимости от направления счета. Значение регистра основания счета (ARR) может вступить в силу мгновенно после его записи в регистр при условии, что установлен флаг $ARRB_EN = 1$ (регистр CNTRL). Если значение предварительного делителя основного счетчика не равно нулю, то счетный регистр делителя будет инкрементироваться по каждому импульсу сигнала TIM_CLK до тех пор, пока не достигнет значения, находящегося в регистре делителя. Далее счетный регистр делителя сбрасывается в ноль, содержимое основного счетчика таймера изменится на 1 и снова начинается счет. Поле DIR определяет, в какую сторону будет меняться значение счетчика: $DIR = 0$ – счетчик считает вверх (Рисунок 35), $DIR = 1$ – счетчик считает вниз (Рисунок 36). Если $CNT_MODE = 00$, то направление счета определяется полем DIR, если $CNT_MODE = 01$, счетчик считает вверх/вниз с автоматическим изменением DIR (Рисунок 37).

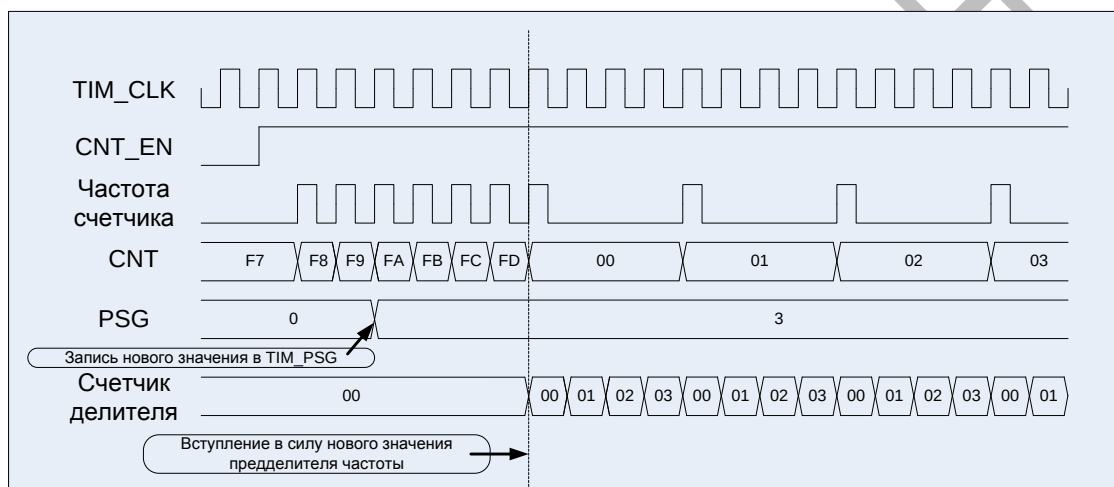


Рисунок 35 – Диаграммы работы счетчика: счет вверх
($CNT_MODE = 00$, $EVENT_SEL = 0000$, $DIR = 0$)

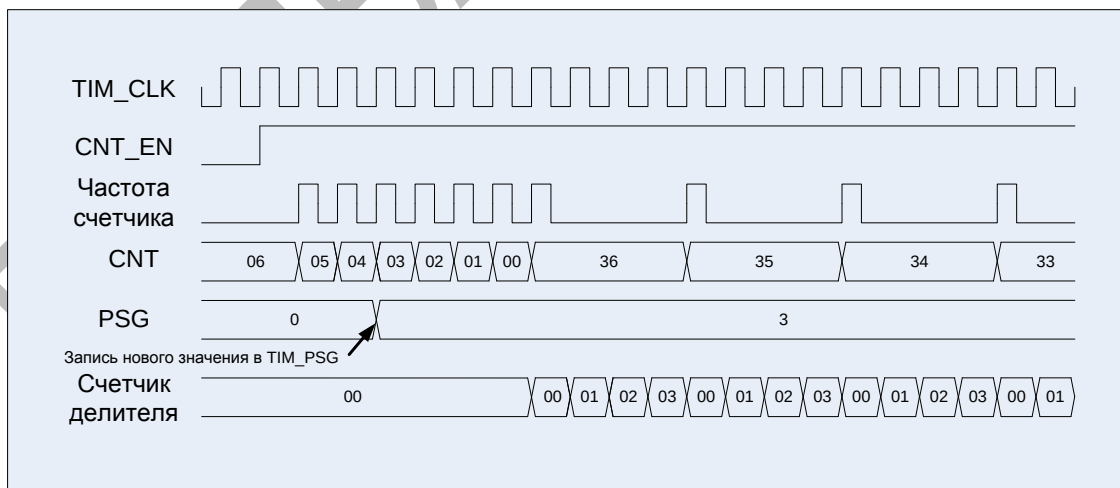


Рисунок 36 – Диаграммы работы счетчика: счет вниз
($CNT_MODE = 00$, $EVENT_SEL = 0000$, $DIR = 1$)

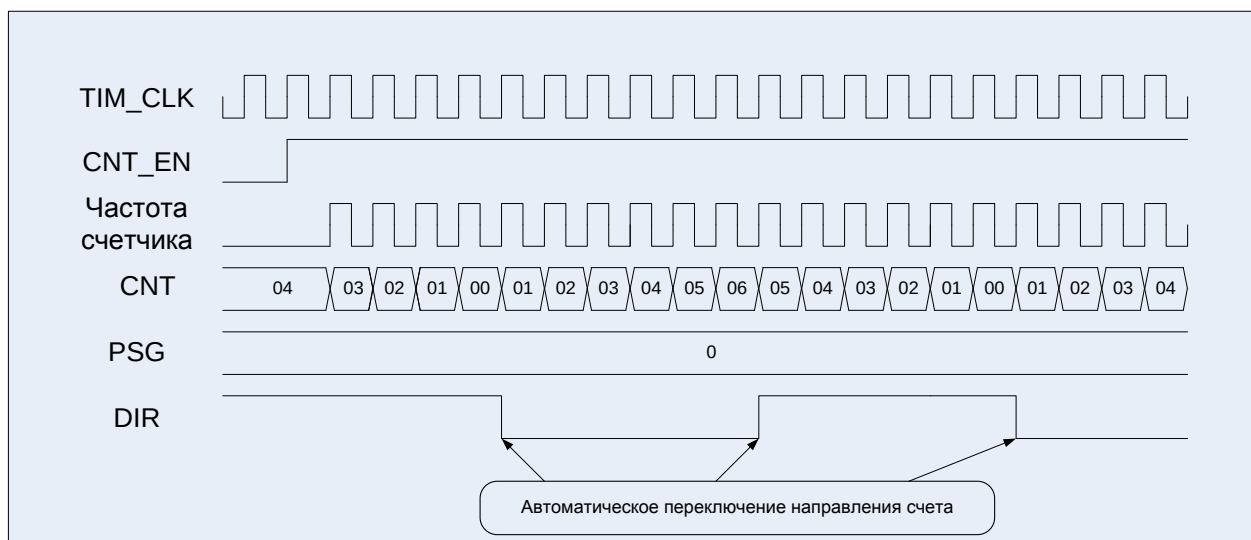


Рисунок 37 – Диаграммы работы счетчика: счет вниз/вверх
(CNT_MODE = 01, EVENT_SEL = 0000, DIR = 1)

События в других счетчиках (CNT==ARR в таймере X)

Каждый из блоков таймеров полностью независим друг от друга, но у них предусмотрена возможность синхронизированной друг с другом работы. Это позволяет создавать более сложные массивы таймеров, которые работают полностью автономно и не требуют написания какого-либо кода программы для выполнения сложных временных функций.

У каждого таймера имеются входы запуска от других трех таймеров, а также внешние входы, связанные с выводами блоков захвата/сравнения.

У каждого из блоков таймеров имеется выход запуска, который соединен с входами других трех таймеров. Синхронизация таймеров возможна в нескольких различных режимах. Ниже показан пример каскадного соединения счетчиков.

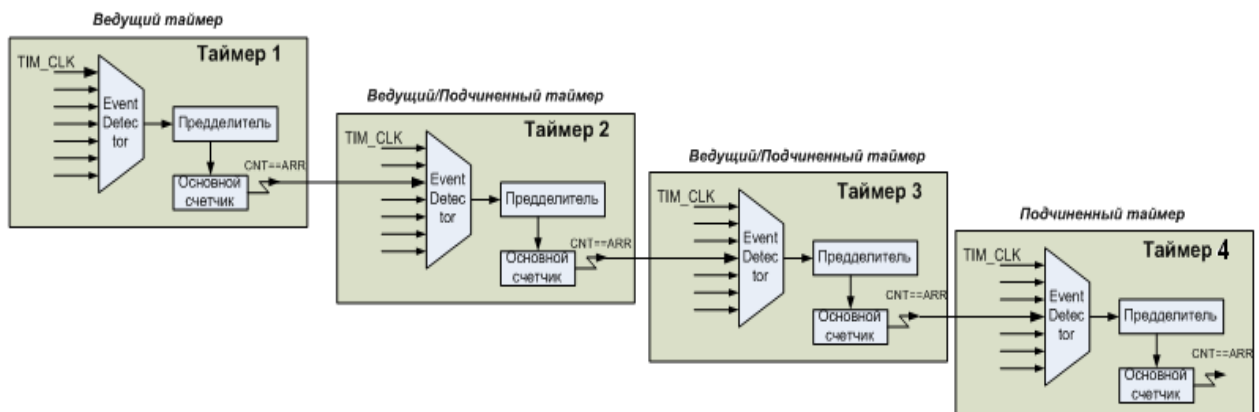


Рисунок 38 – Пример каскадного соединения таймеров

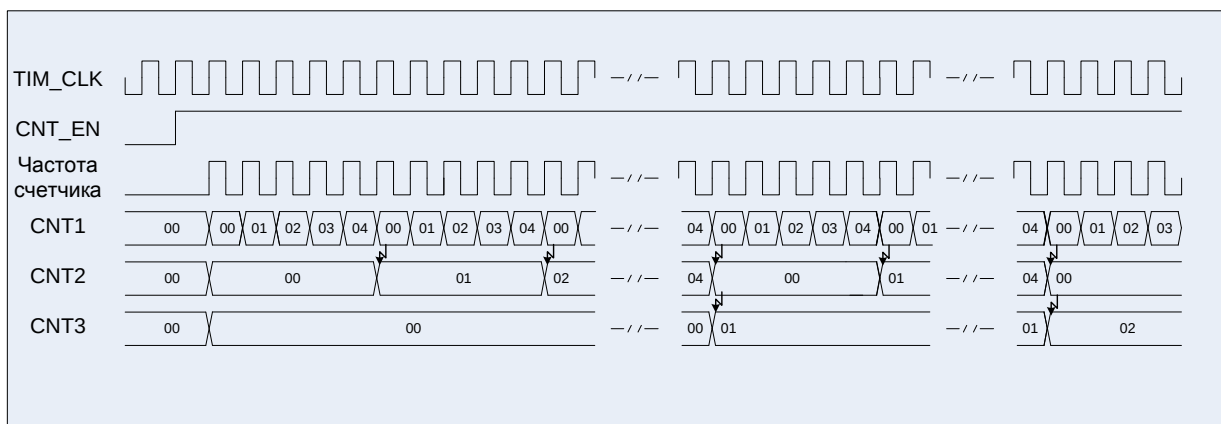


Рисунок 39 – Диаграммы работы трех таймеров в каскаде

DIR_1, DIR_2, DIR_3 = 0;
 EVENT_SEL_1 = 0000, EVENT_SEL_2 = 0001, EVENT_SEL_3 = 0010;
 CNT_MODE_1, CNT_MODE_2, CNT_MODE_3 = 00;

Внешний тактовый сигнал режим 1. События на линиях TxCH0 данного счетчика

Этот режим выбирается, когда EVENT_SEL = 01xx в регистре CNTRL. Счетчик считает только по переднему фронту (см. Рисунок 40). Биты CHSEL[1:0] регистра MDR_TIMERx->CHy_CNTRL не имеют значения. Канал таймера работает с битами CHSEL[1:0] только в режиме захвата, генерируя прерывание по соответствующему фронту.

На входе сигнала стоит фильтр, с помощью которого можно контролировать длительность сигнала, для фильтрации можно использовать как сигнал TIM_CLK, при этом может быть идентифицированная длительность 1, 2, 4, 8 TIM_CLK, также можно при фильтровании использовать производную от TIM_CLK частоту FDTS. Частота семплирования данных задается в регистре CNTRL в поле FDTS.

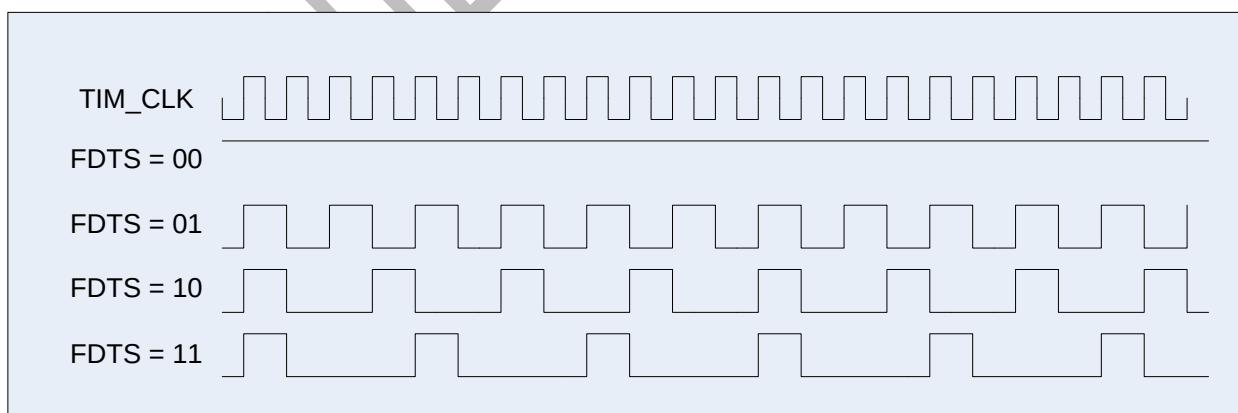


Рисунок 40 – Диаграммы возможных частот семплирования данных (FDTS)

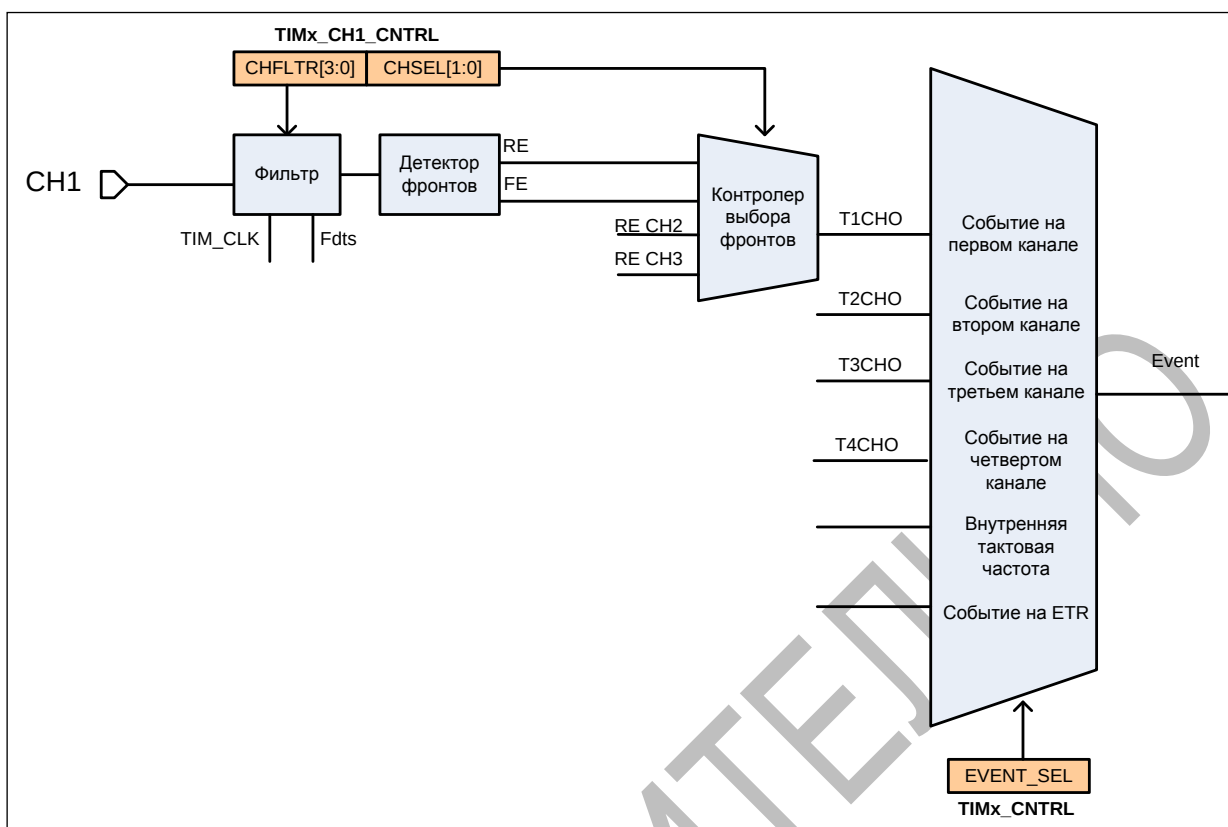


Рисунок 41 – Тактирование с входа первого канала

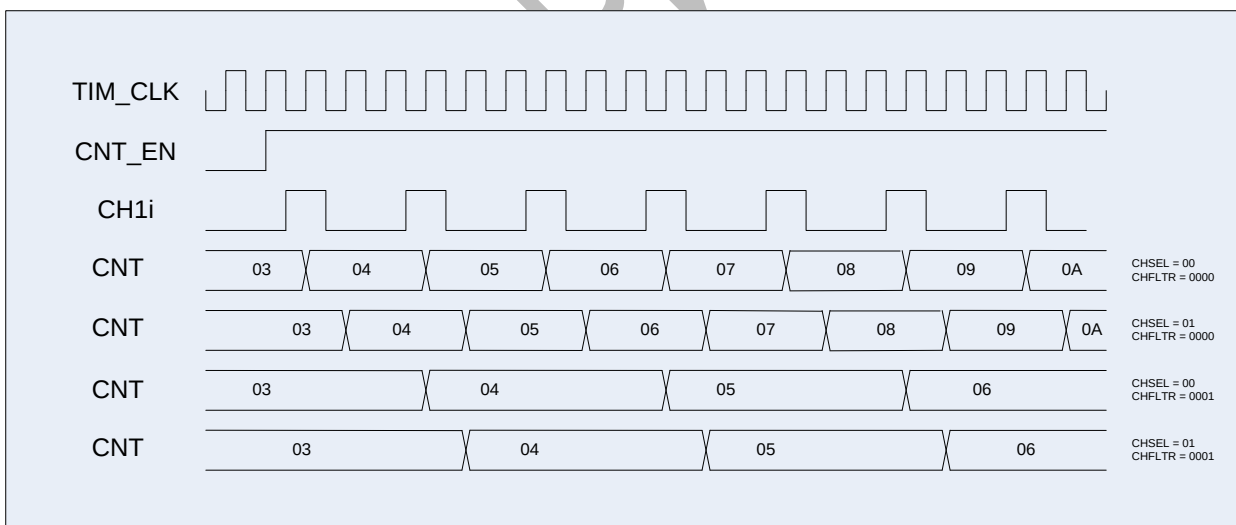


Рисунок 42 – Диаграмма внешнего тактирования с разными вариантами фильтра

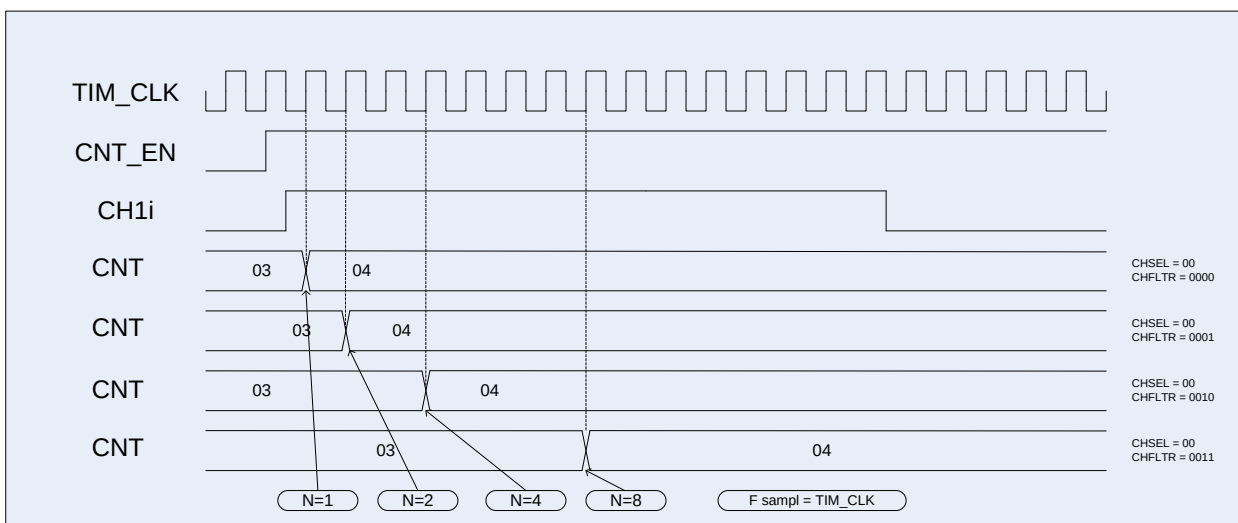


Рисунок 43 – Диаграмма внешнего тактирования с разными вариантами фильтра

Внешний тактовый сигнал режим 2. События на входе ETR данного счетчика

Этот режим выбирается, когда EVENT_SEL = 1000 в регистре CNTRL. В регистре BRKETR_CNTRL можно настроить коэффициент деления 2, 4 или 8 (ETRPSC) данного входа тактовой частоты, а также использовать инверсию входа.

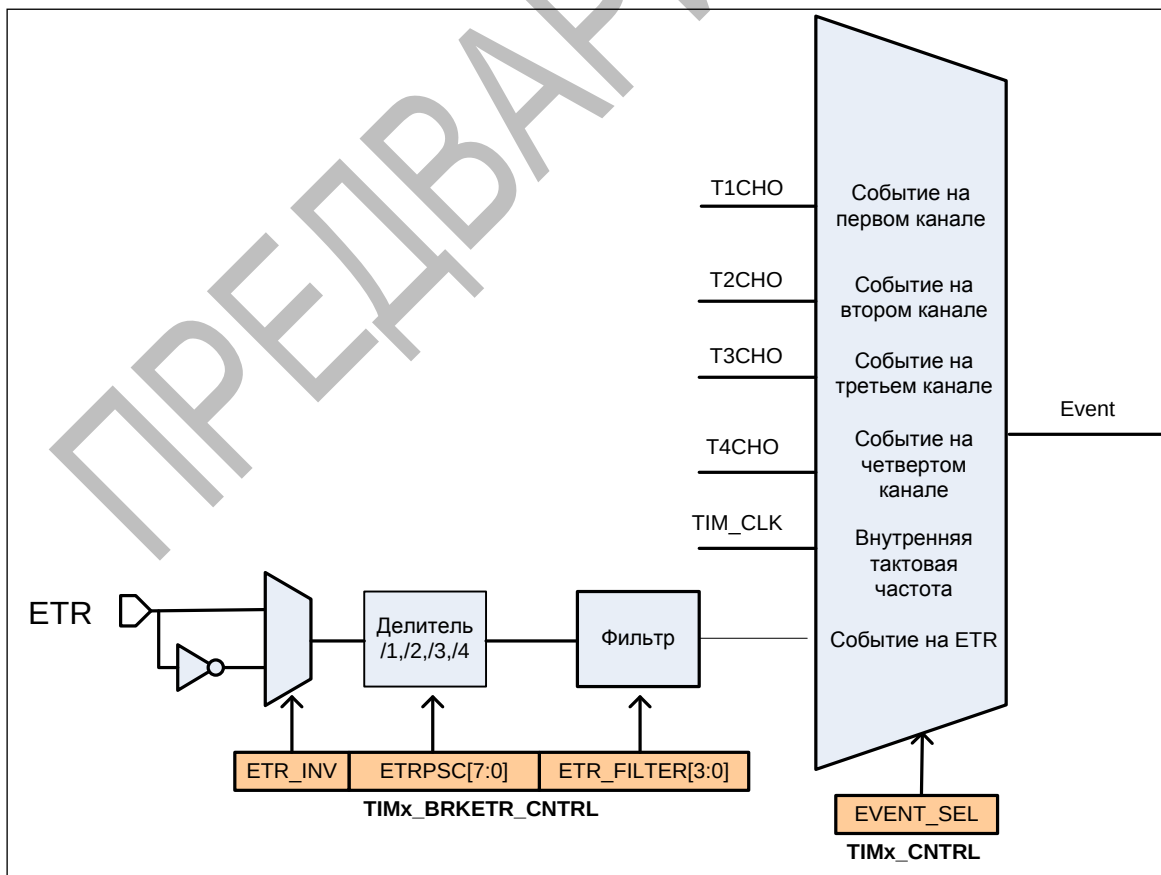


Рисунок 44 – Схема тактирования сигналом со входа ETR

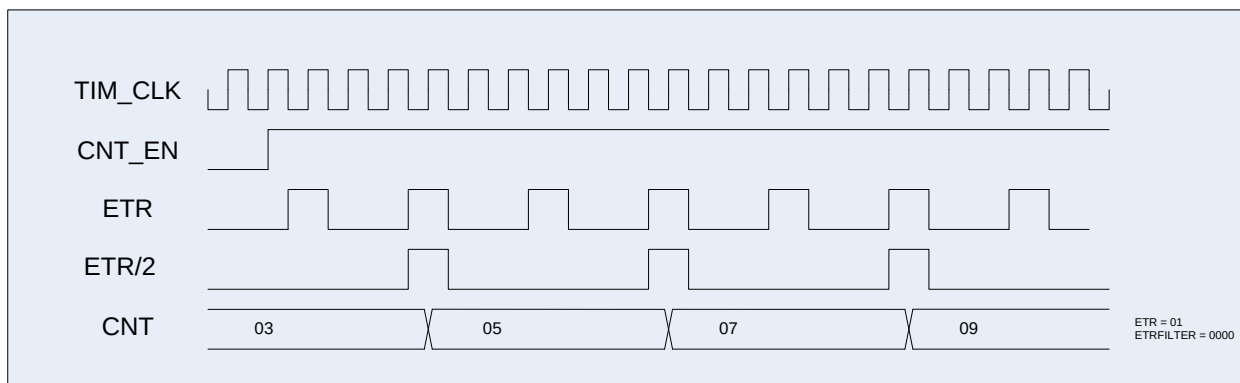


Рисунок 45 – Диаграмма тактирования сигналом со входа ETR

1.4 Режим захвата

Структурная схема блока захвата представлена на рисунке 46.

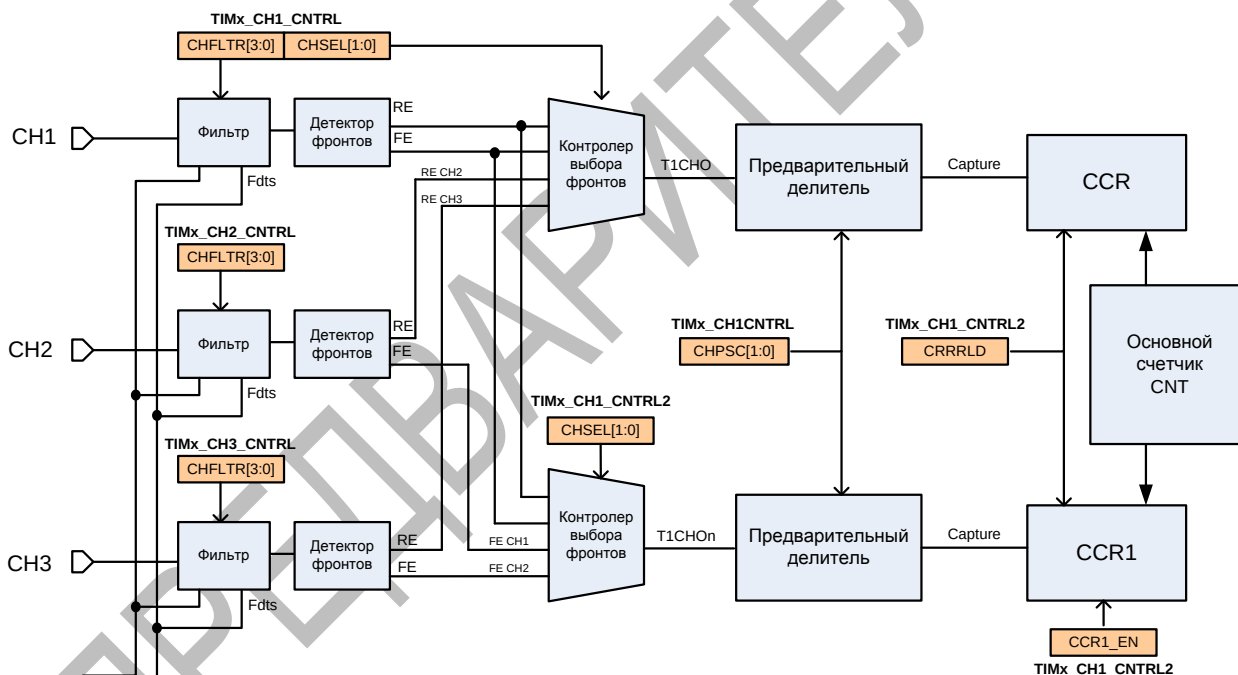


Рисунок 46 – Структурная схема блока захвата на примере канала 1

Для включения режима захвата для определенного канала необходимо в регистре управления каналом CHy_CNTRL записать 1 в поле CAPnPWM. Для регистрации событий по линии Chxi используется схема регистрации событий. Входной сигнал фиксируется в Таймере с частотой Fdts, или TIM_CLK. Также вход может быть настроен на прием импульсов заданной длины за счет конфигурирования блока FILTER. На выходе блока Фильтр вырабатывается сигнал положительного перепада и отрицательного перепада. На

блоке MUX производится выбор используемого для Захвата сигнала, между положительным фронтом канала, отрицательным фронтом канала и положительными и отрицательными фронтами сигналов от других каналов. После блока MUX предварительный делитель может быть использован для фиксации каждого события, каждого второго, каждого четвертого и каждого восьмого события. Выход предварительного делителя является сигналом Capture для регистра CCR, и Capture1 для регистра CCR1 при этом в регистры CCR и CCR1 записывается текущее значение основного счетчика CNT.

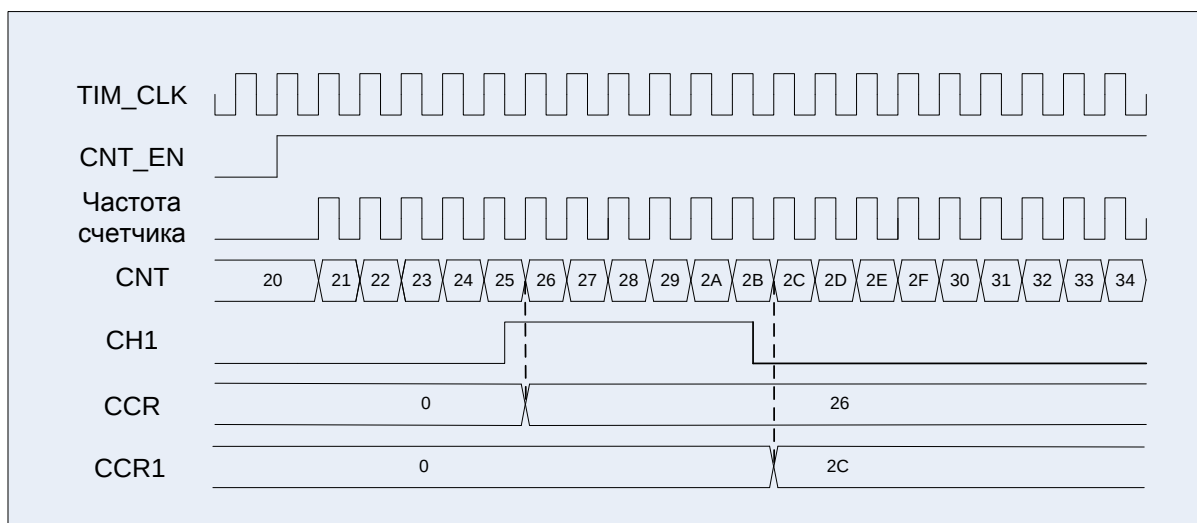


Рисунок 47 – Диаграмма захвата события со входа первого канала

На рисунке 47 показан пример захвата значения основного счетчика в регистр CCR по положительному фронту на входе канала, а в регистр CCR1 по отрицательному фронту на входе канала. В регистре IE можно разрешить выработку прерываний по событию захвата на определенном канале, а в регистре DMA_RE можно разрешить формирование запросов DMA.

1.5 Режим ШИМ

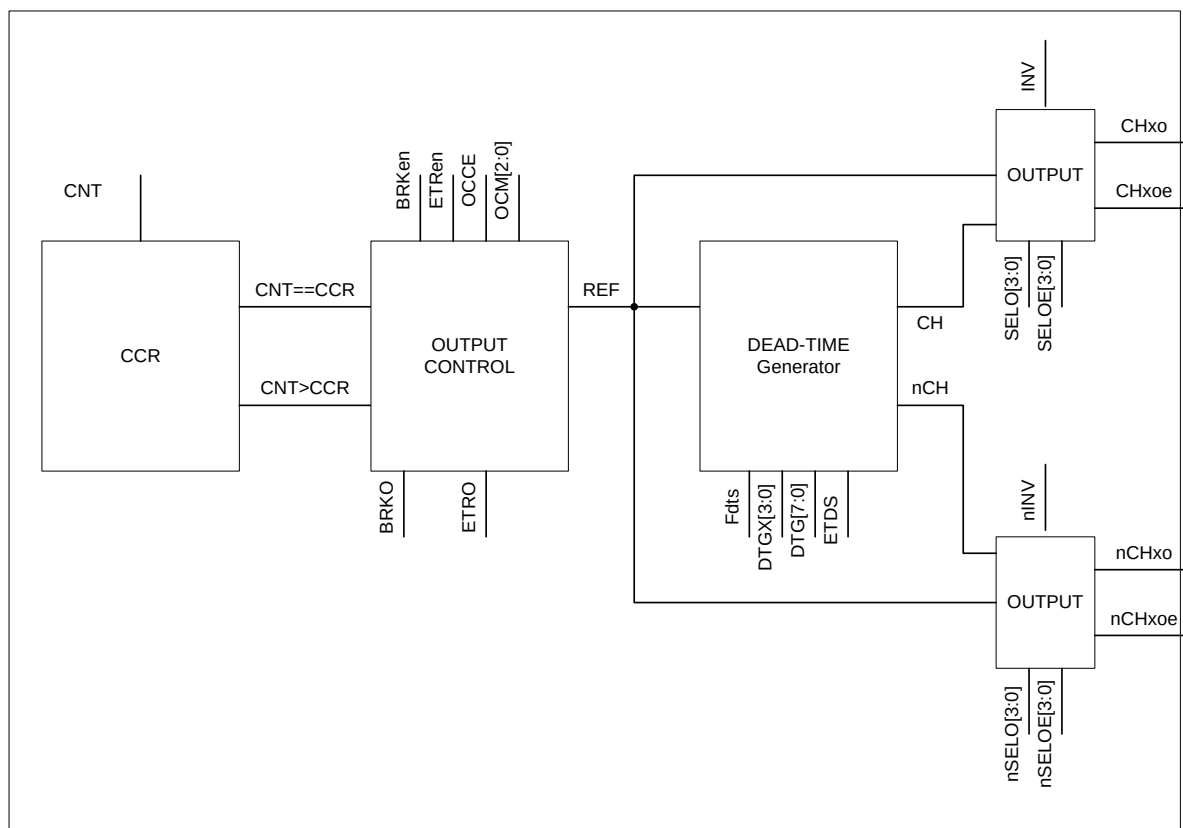


Рисунок 48 – Структурная схема блока сравнения

Для включения режима сравнения для определенного канала необходимо в регистре управления каналом **CHy_CNTRL** записать 0 в поле **CAPnPWM**. При работе в режиме ШИМ выходной сигнал может формироваться на основании сравнения значения в регистре **CCR** и основного счетчика **CNT** или регистров **CCR**, **CCR1** и значения основного счетчика **CNT**. Полученный сигнал может без изменения выдаваться на выходы **ChxO** и **nCHxO**. Либо с применением схемы **DEAD TIME Generator** формируются управляющие сигналы с мертвой зоной. У каждого канала есть два выхода: прямой и инверсный. Для каждого выхода формируется как сигнал для выдачи, так и сигнал разрешения выдачи, т.е. если выход канала должен всегда выдавать тот или иной уровень, то на выходе разрешения выдачи **ChxOE** (для прямого) и на **CHxNOE** (для инверсного) должны формироваться «1». Если канал работает на вход (например, режим захвата), то там всегда должен быть «0» для прямого канала. Сигналы **OE** работают по тем же принципам, что и просто выходные уровни, но у них есть собственные сигналы разрешения вывода **SELOE** и **nSELOE**, в которых можно выбрать постоянный уровень, либо формируемый на основании **REF**.

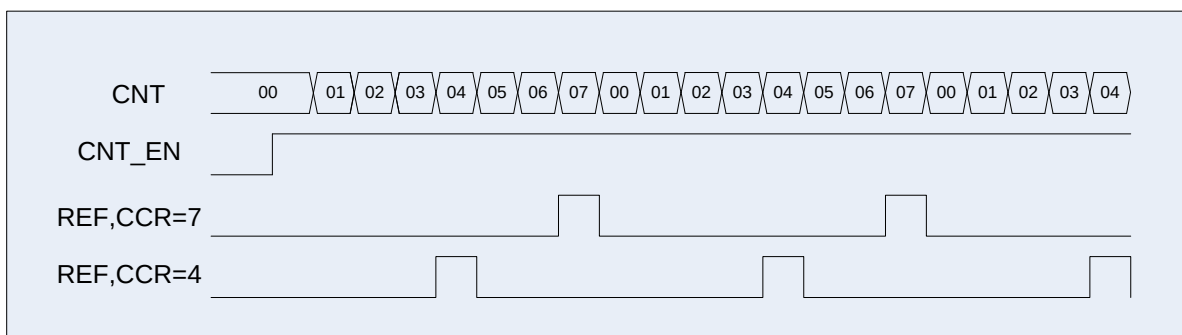


Рисунок 49 – Диаграмма работы схемы в режиме ШИМ, CCR1_EN=0

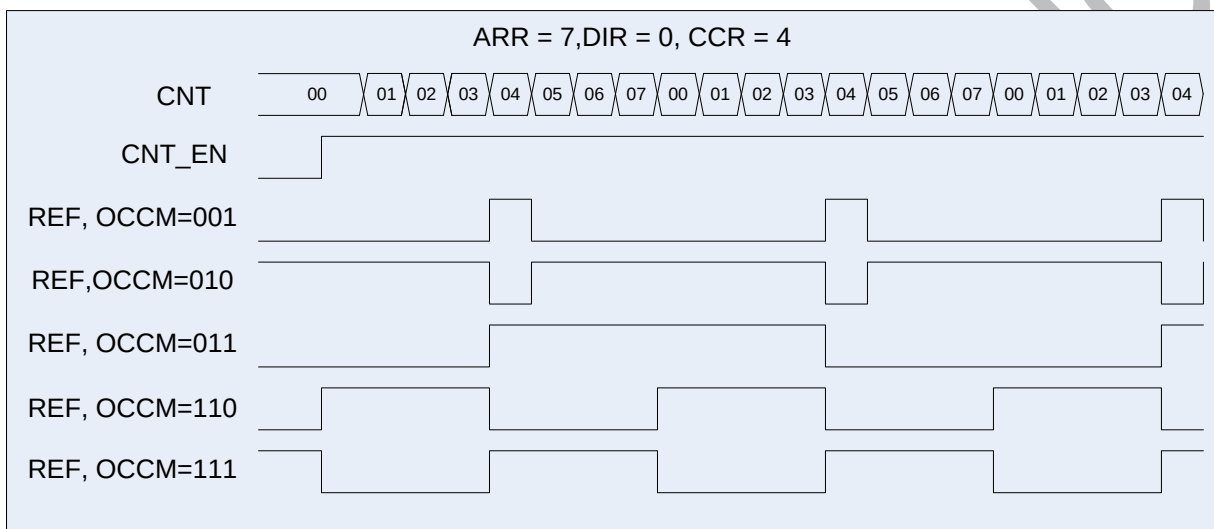


Рисунок 50 – Диаграмма работы схемы в режиме ШИМ, CCR1_EN=0

Сигнал REF может быть очищен с использованием внешнего сигнала с входа ETR или внешнего триггерированного по PCLK сигнала с входа BRK.

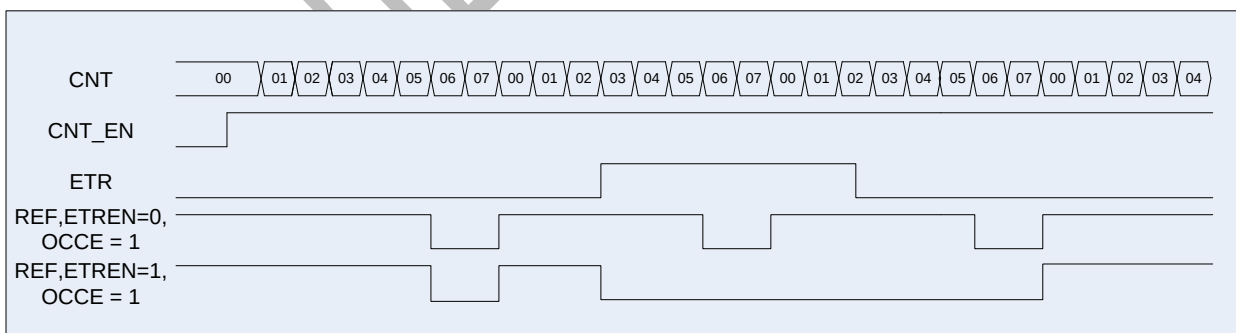


Рисунок 51 – Диаграмма работы схемы в режиме ШИМ, CCR1_EN = 0

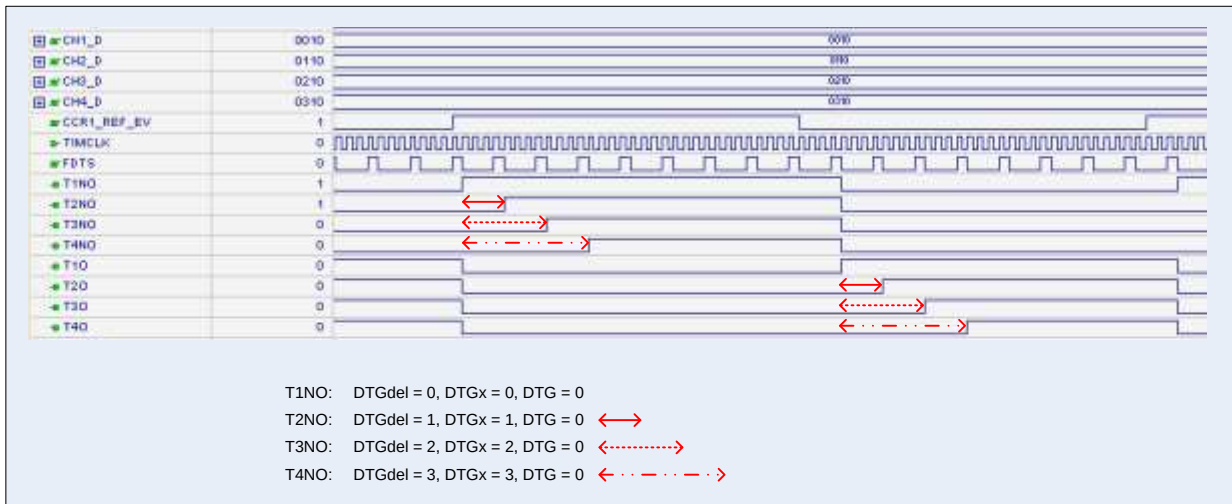


Рисунок 52 – Диаграмма работы схемы DTG

Если $CCR1_EN = 1$, тогда значение основного счетчика CNT сравнивается со значениями регистров CCR и CCR1, и в зависимости от запрограммированного формата выработки сигнала REF (регистры управления каналами таймера CHy_CNTRL поле OCCM) будет формироваться сигнал соответствующей формы.

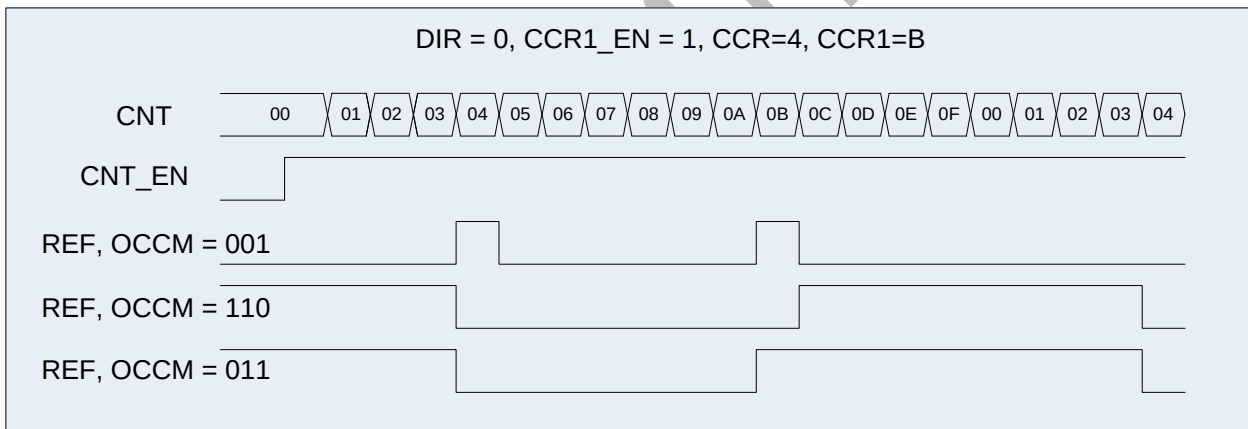


Рисунок 53 – Диаграмма работы схемы в режиме ШИМ, CCR1_EN = 1

При записи новых значений CCR и CCR1, если установлен бит CRRRLD, то регистры CCR1 и CCR получают новые значения только при $CNT = 0$, иначе запись осуществляется немедленно. Факт окончания записи обозначается взведением флага WR_CMPL.

1.6 Примеры

Обычный счетчик

```

RST_CLK->PER_CLOCK = 0xFFFFFFFF;
RST_CLK->TIM_CLOCK = 0x07000000;
TIMERx->CNTRL = 0x00000000;
//Настраиваем работу основного счетчика
TIMERx->CNT = 0x00000000; //Начальное значение счетчика
  
```

TIMERx->PSG = 0x00000000; //Предделитель частоты

TIMERx->ARR = 0x0000000F; //Основание счета

TIMERx->IE = 0x00000002; //Разрешение генерировать прерывание при CNT = ARR

TIMERx->CNTRL = 0x00000001; //Счет вверх по TIM_CLK. Разрешение работы таймера.

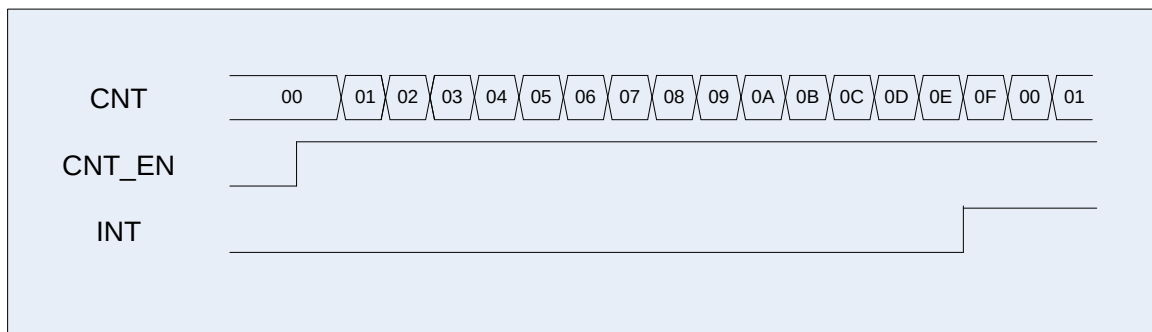


Рисунок 54 – Режим обычного счетчика

Режим захвата

RST_CLK->PER_CLOCK = 0xFFFFFFFF; //Разрешение тактовой частоты таймеров

RST_CLK->TIM_CLOCK = 0x07000000; //Включение тактовой частоты таймеров

TIMERx->CNTRL = 0x00000000; //Режим инициализации таймера

//Настраиваем работу основного счетчика

TIMERx->CNT = 0x00000000; //Начальное значение счетчика

TIMERx->PSG = 0x00000000; //Предделитель частоты

TIMERx->ARR = 0x000000FF; //Основание счета

TIMERx->IE = 0x00001E00; //Разрешение генерировать прерывание

//по переднему фронту на выходе CAP по всем каналам

//Режим работы каналов – захват

TIMERx->Chy_CNTRL[0] = 0x00008000;

TIMERx->Chy_CNTRL[1] = 0x00008002;

TIMERx->Chy_CNTRL[2] = 0x00008001;

TIMERx->Chy_CNTRL[3] = 0x00008003;

//Режим работы выхода канала – канал на выход не работает

TIMERx->Chy_CNTRL1[0] = 0x00000000;

TIMERx->Chy_CNTRL1[1] = 0x00000000;

TIMERx->Chy_CNTRL1[2] = 0x00000000;

TIMERx->Chy_CNTRL1[3] = 0x00000000;

TIMERx->CNTRL = 0x00000001; //Счет вверх по TIM_CLK. Разрешение работы таймера.

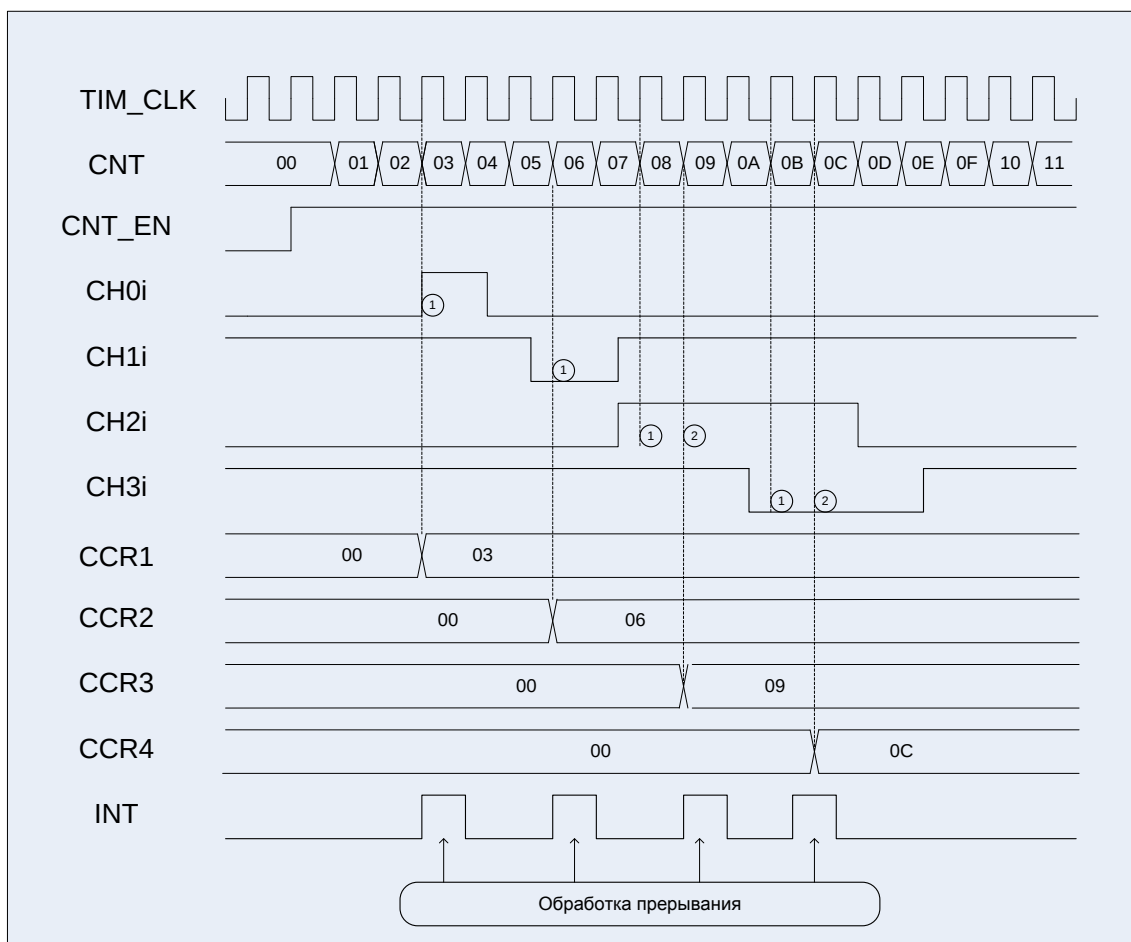


Рисунок 55 – Диаграммы примера работы в режиме захвата

Режим ШИМ

```

RST_CLK->PER_CLOCK = 0xFFFFFFFF; //Разрешение тактовой частоты таймеров
RST_CLK->TIM_CLOCK = 0x07000000; //Включение тактовой частоты таймеров
TIMERx->CNTRL = 0x00000000; //Режим инициализации таймера
//Настраиваем работу основного счетчика
TIMERx->CNT = 0x00000000; //Начальное значение счетчика
TIMERx->PSG = 0x00000000; //Предделитель частоты
TIMERx->ARR = 0x00000010; //Основание счета

TIMERx->IE = 0x000001E0; //Разрешение генерировать прерывание
//по переднему фронту на выходе REF по всем каналам
//Режим работы каналов – ШИМ
TIMERx->Chy_CNTRL[0] = 0x00000200;
TIMERx->Chy_CNTRL[1] = 0x00000200;
TIMERx->Chy_CNTRL[2] = 0x00000400;
TIMERx->Chy_CNTRL[3] = 0x00000600;

//Режим работы выхода канала – канал на выход не работает
TIMERx->Chy_CNTRL1[0] = 0x00000099;
TIMERx->Chy_CNTRL1[1] = 0x00000099;
TIMERx->Chy_CNTRL1[2] = 0x00000099;
TIMERx->Chy_CNTRL1[3] = 0x00000099;

```

//Разрешение работы таймера.
 TIMEx->CNTRL = 0x00000001; //Счет вверх по TIM_CLK.

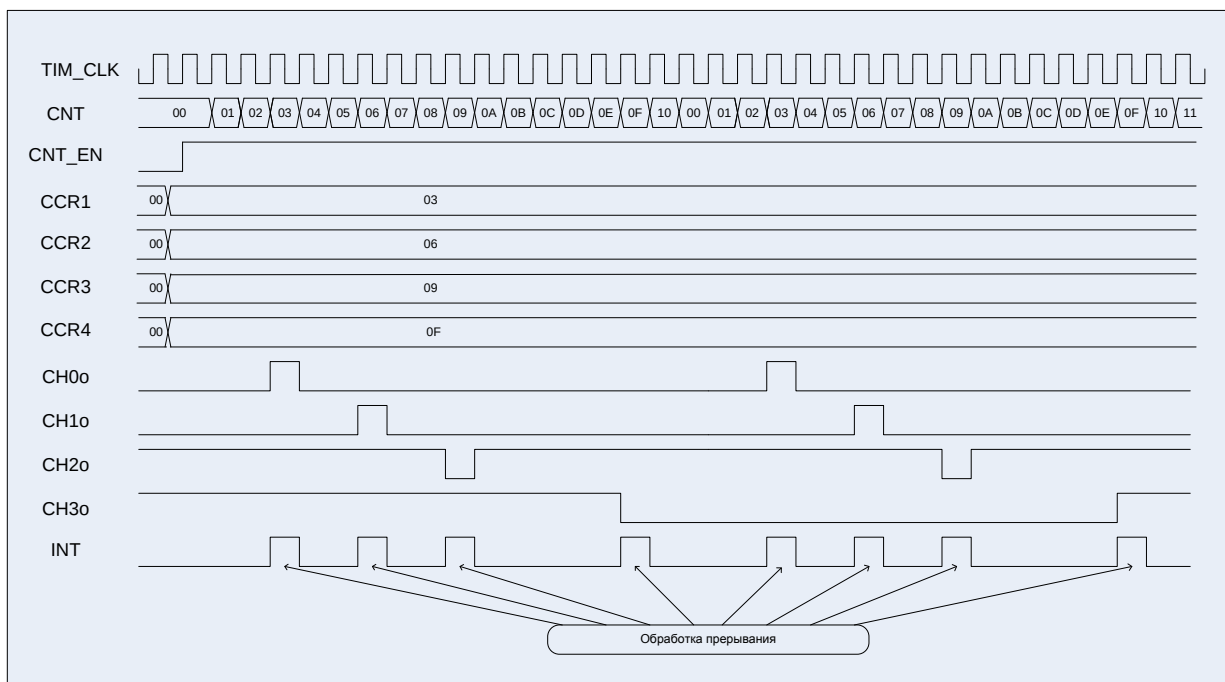


Рисунок 56 – Диаграммы примера работы в режиме ШИМ

1.7 Описание регистров блока таймера

Таблица 291– Базовые адреса и смещения регистров управления таймера

Адрес	Название	Описание
0x4007_0000	Timer1	Контроллер Timer1
0x4007_8000	Timer2	Контроллер Timer2
0x400D_8000	Timer3	Контроллер Timer3
0x400E_0000	Timer4	Контроллер Timer4
Смещение		
0x00	CNT[31:0]	Основной счетчик таймера
0x04	PSG[15:0]	Делитель частоты при счете основного счетчика
0x08	ARR[31:0]	Основание счета основного счетчика
0x0C	CNTRL[11:0]	Регистр управления основного счетчика
0x10	CCR1[31:0]	Регистр сравнения, захвата для 1 канала таймера
0x14	CCR2[31:0]	Регистр сравнения, захвата для 2 канала таймера
0x18	CCR3[31:0]	Регистр сравнения, захвата для 3 канала таймера
0x1C	CCR4[31:0]	Регистр сравнения, захвата для 4 канала таймера
0x20	CH1_CNTRL[15:0]	Регистр управления для 1 канала таймера
0x24	CH2_CNTRL[15:0]	Регистр управления для 2 канала таймера
0x28	CH3_CNTRL[15:0]	Регистр управления для 3 канала таймера
0x2C	CH4_CNTRL[15:0]	Регистр управления для 4 канала таймера
0x30	CH1_CNTRL1[15:0]	Регистр управления 1 для 1 канала таймера
0x34	CH2_CNTRL1[15:0]	Регистр управления 1 для 2 канала таймера
0x38	CH3_CNTRL1[15:0]	Регистр управления 1 для 3 канала таймера
0x3C	CH4_CNTRL1[15:0]	Регистр управления 1 для 4 канала таймера
0x40	CH1_DTG[15:0]	Регистр управления DTG для 1 канала таймера
0x44	CH2_DTG[15:0]	Регистр управления DTG для 2 канала таймера
0x48	CH3_DTG[15:0]	Регистр управления DTG для 3 канала таймера
0x4C	CH4_DTG[15:0]	Регистр управления DTG для 4 канала таймера
0x50	BRKETR_CNTRL[15:0]	Регистр управления входом BRK и ETR
0x54	STATUS[15:0]	Регистр статуса таймера
0x58	IE[15:0]	Регистр разрешения прерывания таймера
0x5C	DMA_RE[15:0]	Регистр разрешения запросов DMA от прерываний таймера
0x60	CH1_CNTRL2[15:0]	Регистр управления 2 для 1 канала таймера
0x64	CH2_CNTRL2[15:0]	Регистр управления 2 для 2 канала таймера
0x68	CH3_CNTRL2[15:0]	Регистр управления 2 для 3 канала таймера
0x6C	CH4_CNTRL2[15:0]	Регистр управления 2 для 4 канала таймера
0x70	CCR11[31:0]	Регистр сравнения, захвата 1 для 1 канала таймера
0x74	CCR21[31:0]	Регистр сравнения, захвата 1 для 2 канала таймера
0x78	CCR31[31:0]	Регистр сравнения, захвата 1 для 3 канала таймера
0x7C	CCR41[31:0]	Регистр сравнения, захвата 1 для 4 канала таймера
0x80	DMA_RE1[15:0]	Регистр разрешения запросов DMA от прерываний канала 1 таймера
0x84	DMA_RE2[15:0]	Регистр разрешения запросов DMA от прерываний канала 2 таймера
0x88	DMA_RE3[15:0]	Регистр разрешения запросов DMA от прерываний канала 3 таймера
0x8C	DMA_RE4[15:0]	Регистр разрешения запросов DMA от прерываний канала 4 таймера

1.7.1 CNT

Таблица 292 – Основной счетчик таймера CNT

Номер	31...0
Доступ	R/W
Сброс	0
	CNT[31:0]

Таблица 293 – Описание бит регистра CNT

№	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	CNT[31:0]	Значение основного счетчика таймера

1.7.2 PSG

Таблица 294 – Делитель частоты при счете основного счетчика PSG

Номер	31..16	15..0
Доступ	R/W	R/W
Сброс	0	0
	-	PSG[15:0]

Таблица 295 – Описание бит регистра PSG

№	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...16	-	Зарезервировано
15...0	PSG[15:0]	Значение предварительного делителя счетчика Основной счетчик считает на частоте $CLK = TIM_CLK / (PSG + 1)$

1.7.3 ARR

Таблица 296 – Основание счета основного счетчика ARR

Номер	31...0
Доступ	R/W
Сброс	0
	ARR[31:0]

Таблица 297 – Описание бит регистра ARR

№	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	ARR[31:0]	Основание счета для основного счетчика $CNT = [0...ARR]$

1.7.4 CNTRL

Таблица 298 – Регистр управления основного счетчика CNTRL

Номер	31...12	11...8	7...6	5...4	3	2	1	0
Доступ	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Сброс	0	0000	00	00	0	0	0	0
	-	EVENT SEL [3:0]	CNT MODE [1:0]	FDTs [1:0]	DIR	WR CMPL	ARRB EN	CNT EN

Таблица 299 – Описание бит регистра CNTRL

№	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений.
31...12	-	Зарезервировано
11...8	EVENT_SEL [3:0]	Биты выбора источника событий 4'b0000 – событие переднего фронта на TIM_CLK 4'b0001 – CNT == ARR в таймере 1 4'b0010 – CNT == ARR в таймере 2 4'b0011 – CNT == ARR в таймере 3 4'b0100 – событие на первом канале 4'b0101 – событие на втором канале 4'b0110 – событие на третьем канале 4'b0111 – событие на четвертом канале 4'b1000 – событие переднего фронта ETR 4'b1001 – событие заднего фронта ETR 4'b1010 – CNT == ARR в таймере 4
7..6	CNT_MODE [1:0]	Режим счета основного счетчика 2'b00 – счетчик вверх при DIR=0 счетчик вниз при DIR=1 при PSG = 0 2'b01 – счетчик вверх/вниз с автоматическим изменением DIR при CNT = 0 или CNT = ARR 2'b1x – счетчик вверх при DIR=0 счетчик вниз при DIR=1 Режимы CNT_MODE[1:0] = 2'b00 и CNT_MODE[1:0] = 2'b01 действуют, если EVENT_SEL = 4'b0000; режим CNT_MODE[1:0] = 2'b1x действует, если EVENT_SEL != 4'b0000.
5...4	FDTs[1:0]	Частота семплирования данных FDTs 2'b00 – каждый TIM_CLK 2'b01 – каждый второй TIM_CLK 2'b10 – каждый третий TIM_CLK 2'b11 – каждый четвертый TIM_CLK
3	DIR	Направление счета основного счетчика 0 – вверх, от 0 до ARR 1 – вниз, от ARR до 0
2	WR_CMPL	Окончание записи, при задании нового значения регистров CNT, PSG и ARR 1 – данные не записаны и идет запись 0 – новые данные можно записывать
1	ARRB_EN	Разрешение мгновенного обновления ARR 0 – ARR будет перезаписан в момент записи в ARR 1 – ARR будет перезаписан при завершении счета CNT
0	CNT_EN	Разрешение работы таймера 0 – таймер отключен

		1 – таймер включен
--	--	--------------------

ПРЕДВАРИТЕЛЬНО

1.7.5 CCRy

Таблица 300 – Регистр сравнения/захвата для 'у' канала таймера CCRy

Номер	31...0
Доступ	R/W
Сброс	0
	CCR[31:0]

Таблица 301 – Описание бит регистра CCRy

№	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	CCR[31:0]	Значение CCR, с которым сравнивается CNT при работе в ШИМ режиме. Значение CNT, при котором произошел факт захвата события, в режиме захвата

1.7.6 CCRy1

Таблица 302 – Регистр сравнения/захвата для 'у' канала таймера CCRy1

Номер	31...0
Доступ	R/W
Сброс	0
	CCR1[31:0]

Таблица 303 – Описание бит регистра CCRy1

№	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	CCR1[31:0]	Значение CCR1, с которым сравнивается CNT при работе в ШИМ режиме. Значение CNT, при котором произошел факт захвата события, в режиме захвата

1.7.7 Chy_CNTRL

Таблица 304 – Регистр управления для 'у' канала таймера Chy_CNTRL

Номер	31...16	15
Доступ	U	R/W
Сброс	0	0
	-	CAP nPWM

Номер	14	13	12	11...9	8	7...6	5...4	3...0
Доступ	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Сброс	0	0	0	000	0	00	00	0000
	WR CMPL	ETREN	BRKEN	OCCM [2:0]	OCCE	CHPSC [1:0]	CHSEL [1:0]	CHFLT R[3:0]

Таблица 305 – Описание бит регистра Chy_CNTRL

№	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...16	-	Зарезервировано

15	CAP nPWM	Режим работы канала Захват или ШИМ 1 – канал работает в режиме Захват 0 – канал работает в режиме ШИМ
14	WR CMPL	Флаг окончания записи, при задании нового значения регистра CCR 1 – данные не записаны и идет запись 0 – новые данные можно записывать
13	ETREN	Разрешения сброса по выводу ETR 0 – запрещен сброс 1 – разрешен
12	BRKEN	Разрешение сброса по выводу BRK 0 – запрещен сброс 1 – разрешен
11...9	OCCM[2:0]	Формат выработки сигнала REF в режиме ШИМ Если CCR1_EN = 0: 000 – всегда 0 001 – 1, если CNT = CCR; 010 – 0, если CNT = CCR; 011 – переключение REF, если CNT = CCR; 100 – всегда 0; 101 – всегда 1; 110 – 1, если DIR= 0 (счет вверх), CNT<CCR, иначе 0; 0, если DIR= 1 (счет вниз), CNT>CCR, иначе 1; 111 – 0, если DIR= 0 (счет вверх), CNT<CCR, иначе 1; 1, если DIR= 1 (счет вниз), CNT>CCR, иначе 0. Если CCR1_EN = 1: 000 – всегда 0; 001 – 1, если CNT = CCR или CNT = CCR1 010 – 0, если CNT = CCR или CNT = CCR1; 011 – переключение REF, если CNT =CCR или CNT =CCR1; 100 – всегда 0; 101 – всегда 1; 110 – 1, если DIR = 0 (счет вверх), CCR< CNT< CCR1, иначе 0; 0, если DIR = 1 (счет вниз), CCR < CNT < CCR1, иначе 1; 111 – 0, если DIR = 0 (счет вверх), CCR< CNT < CCR1, иначе 1; 1, если DIR = 1 (счет вниз), CCR< CNT< CCR1, иначе 0. При условии, что CCR < CCR1
8	OCCE	Разрешение работы ETR 0 – запрет ETR 1 – разрешение ETR
7...6	CHPSC[1:0]	Предварительный делитель входного канала 00 – нет деления 01 – /2 10 – /4 11 – /8
5...4	CHSEL[1:0]	Выбор события по входному каналу CHxі для фиксации значения основного счетчика (регистр MDR_TIMERx->CNT) в регистр CCR: 00 – положительный фронт на входном канале CHxі; 01 – отрицательный фронт на входном канале CHxі; 10 – положительный фронт от других каналов Для первого канала от 2 канала Для второго канала от 3 канала Для третьего канала от 4 канала Для четвертого канала от 1 канала 11 – положительный фронт от других каналов Для первого канала от 3 канала Для второго канала от 4 канала Для третьего канала от 1 канала

		Для четвертого канала от 2 канала
3...0	CHFLTR[3:0]	Сигнал зафиксирован: 0000 – в 1 триггере на частоте TIM_CLK 0001 – в 2 триггерах на частоте TIM_CLK 0010 – в 4 триггерах на частоте TIM_CLK 0011 – в 8 триггерах на частоте TIM_CLK 0100 – в 6 триггерах на частоте FDTS/2 0101 – в 8 триггерах на частоте FDTS/2 0110 – в 6 триггерах на частоте FDTS/4 0111 – в 8 триггерах на частоте FDTS/4 1000 – в 6 триггерах на частоте FDTS/8 1001 – в 8 триггерах на частоте FDTS/8 1010 – в 5 триггерах на частоте FDTS/16 1011 – в 6 триггерах на частоте FDTS/16 1100 – в 8 триггерах на частоте FDTS/16 1101 – в 5 триггерах на частоте FDTS/32 1110 – в 6 триггерах на частоте FDTS/32 1111 – в 8 триггерах на частоте FDTS/32

1.7.8 CHy_CNTRL1

Таблица 306 – Регистр управления 1 для 'у' канала таймера Chy_CNTRL1

Номер	31...13	12	11...10	9...8	7...5	4	3...2	1...0
Доступ	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Сброс	0	0	00	00	0	0	00	00
	-	NINV	NSELO [1:0]	NSELOE [1:0]	-	INV	SELO [1:0]	SELOE [1:0]

Таблица 307 – Описание бит регистра Chy_CNTRL1

№	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений.
31...13	-	Зарезервировано
12	NINV	Режим выходной инверсии инверсного канала 0 – выход не инвертируется 1 – выход инвертируется
11...10	NSELO[1:0]	Режим работы выхода инверсного канала 00 – всегда на выход выдается 0, канал на выход не работает 01 – всегда на выход выдается 1, канал всегда работает на выход 10 – на выход выдается сигнал REF. 11 – на выход выдается сигнал с DTG.
9...8	NSELOE[1:0]	Режим работы инверсного канала на выход 00 – всегда на ОЕ выдается 0, канал на выход не работает 01 – всегда на ОЕ выдается 1, канал всегда работает на выход 10 – на ОЕ выдается сигнал REF, при REF = 0 вход, при REF = 1 выход. 11 – на ОЕ выдается сигнал с DTG, при CHn = 0 вход, при CHn = 1 выход
7...5	-	Зарезервировано
4	INV	Режим выходной инверсии прямого канала 0 – выход не инвертируется 1 – выход инвертируется
3...2	SELO[1:0]	Режим работы выхода прямого канала 00 – всегда на выход выдается 0, канал на выход не работает 01 – всегда на выход выдается 1, канал всегда работает на выход 10 – на выход выдается сигнал REF. 11 – на выход выдается сигнал с DTG.
1...0	SELOE[1:0]	Режим работы прямого канала на выход 00 – всегда на ОЕ выдается 0, канал на выход не работает

		01 – всегда на ОЕ выдается 1, канал всегда работает на выход 10 – на ОЕ выдается сигнал REF, при REF = 0 вход, при REF = 1 выход. 11 – на ОЕ выдается сигнал с DTG, при CH = 0 вход, при CH = 1 выход
--	--	---

1.7.9 CHy_CNTRL2

Таблица 308 – Регистр управления 2 для 'у' канала таймера Chy_CNTRL2

Номер	31...4	4	3	2	1...0
Доступ	U	R/W	R/W	R/W	R/W
Сброс	0	0	0	0	00
	-	EV_DELAY	CCRRLD	CCR1_EN	CHSEL [1:0]

Таблица 309 – Описание бит регистра Chy_CNTRL2

№	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...5	-	Зарезервировано
4	EV_DELAY	Задержка события захвата до обновления регистров CCR и CCR1 0 – событие захвата происходит в момент обнаружения события, асинхронно с обновлением регистров CCR и CCR1 1 – событие захвата происходит синхронно с обновлением информации в регистрах CCR и CCR1
3	CCRRLD	Разрешение обновления регистров CCR и CCR1 0 – обновление возможно в любой момент времени 1 – обновление будет осуществлено только при CNT = 0
2	CCR1_EN	Разрешение работы регистра CCR1 0 – CCR1 не используется 1 – CCR1 используется
1...0	CHSEL1[1:0]	Выбор события по входному каналу CHxi для фиксации значения основного счетчика (регистр MDR_TIMERx->CNT) в регистр CCR1: 00 – положительный фронт на входном канале CHxi; 01 – отрицательный фронт на входном канале CHxi; 10 – отрицательный фронт от других каналов Для первого канала от 2 канала Для второго канала от 3 канала Для третьего канала от 4 канала Для четвертого канала от 1 канала 11 – отрицательный фронт от других каналов Для первого канала от 3 канала Для второго канала от 4 канала Для третьего канала от 1 канала Для четвертого канала от 2 канала

1.7.10 CHy_DTG

Таблица 310 – Регистр CHy_DTG управления DTG

Номер	31..16	15...8	7...5	4	3...0
Доступ	U	R/W	U	R/W	R/W
Сброс	0	00000000	000	0	0000
	-	DTG[7:0]	-	EDTS	DTGx [3:0]

Таблица 311 – Описание бит регистра CHy_DTG

№	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...16	-	Зарезервировано
15...8	DTGx[7:0]	Основной делитель частоты Задержка DTGdel = DTGx*(DTG+1).
7...5	-	Зарезервировано
4	EDTS	Частота работы DTG 0 – TIM_CLK 1 – FDTS
3...0	DTG [3:0]	Предварительный делитель частоты DTG

1.7.11 BRKETR_CNTRL

Таблица 312 – Регистр BRKETR_CNTRL управления входом BRK и ETR

Номер	31...8	7...4	3...2	1	0
Доступ	U	R/W	R/W	R/W	R/W
Сброс		0000	00	0	0
	-	ETR FILTER [3:0]	ETR PSC [1:0]	ETR INV	BRK INV

Таблица 313 – Описание бит регистра BRKETR_CNTRL

№	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...8	-	Зарезервировано
7...4	ETR FILTER[3:0]	Цифровой фильтр на входе ETR. Сигнал зафиксирован: 0000 – в 1 триггере на частоте TIM_CLK 0001 – в 2 триггерах на частоте TIM_CLK 0010 – в 4 триггерах на частоте TIM_CLK 0011 – в 8 триггерах на частоте TIM_CLK 0100 – в 6 триггерах на частоте FDTS/2 0101 – в 8 триггерах на частоте FDTS/2 0110 – в 6 триггерах на частоте FDTS/4 0111 – в 8 триггерах на частоте FDTS/4 1000 – в 6 триггерах на частоте FDTS/8 1001 – в 8 триггерах на частоте FDTS/8 1010 – в 5 триггерах на частоте FDTS/16 1011 – в 6 триггерах на частоте FDTS/16 1100 – в 8 триггерах на частоте FDTS/16 1101 – в 5 триггерах на частоте FDTS/32 1110 – в 6 триггерах на частоте FDTS/32 1111 – в 8 триггерах на частоте FDTS/32
3...2	ETRPSC[1:0]	Асинхронный предделитель внешней частоты 00 – без деления 01 – /2 10 – /4 11 – /8
1	ETR INV	Инверсия входа ETR 0 – без инверсии 1 – инверсия
0	BRK INV	Инверсия входа BRK 0 – без инверсии 1 – инверсия

ПРЕДВАРИТЕЛЬНО

1.7.12 STATUS

Таблица 314 – Регистр статуса таймера STATUS

Номер	31...17	16...13	12...9	8...5
Доступ	U	U	R/W	R/W
Сброс	0	0	0	0
	-	CCR CAP1 EVENT [3:0]	CCR REF EVENT [3:0]	CCR CAP EVENT [3:0]

Номер	4	3	2	1	0
Доступ	R/W	R/W	R/W	R/W	R/W
Сброс	0	0	0	0	0
	BRK EVENT	ETR FE EVENT	ETR RE EVENT	CNT ARR EVENT	CNT ZERO EVENT

Таблица 315 – Описание бит регистра STATUS

№	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...17	-	Зарезервировано
16...13	CCR CAP1 EVENT[3:0]	Событие настроенного фронта на входе CN _{xi} канала таймера 0 – нет события 1 – есть событие Сбрасывается записью 0, если запись одновременно с новым событием, приоритет у нового события. Бит 0 – первый канал Бит 3 – четвертый канал
12...9	CCR REF EVENT[3:0]	Событие переднего фронта на выходе REF каналов таймера 0 – нет события 1 – есть событие Сбрасывается записью 0, если запись одновременно с новым событием, приоритет у нового события. Бит 0 – первый канал Бит 3 – четвертый канал
8...5	CCR CAP EVENT[3:0]	Событие настроенного фронта на входе CN _{xi} канала таймера 0 – нет события 1 – есть событие Сбрасывается записью 0, если запись одновременно с новым событием, приоритет у нового события. Бит 0 – первый канал Бит 3 – четвертый канал
4	BRK EVENT	Триггерированное по PCLK состояние входа BRK, 0 – BRK == 0 1 – BRK == 1 Сбрасывается записью 0, при условии наличия 0 на входе BRK
3	ETR FE EVENT	Событие заднего фронта на входе ETR 0 – нет события 1 – есть событие Сбрасывается записью 0, если запись одновременно с новым событием, приоритет у нового события.

2	ETR RE EVENT	Событие переднего фронта на входе ETR 0 – нет события 1 – есть событие Сбрасывается записью 0, если запись одновременно с новым событием, приоритет у нового события.
1	CNT ARR EVENT	Событие совпадения CNT с ARR 0 – нет события 1 – есть событие Сбрасывается записью 0, если запись одновременно с новым событием совпадения, приоритет у нового события. Если с момента совпадения до момента программного сброса CNT и ARR не изменили состояния, то флаг повторно не взводится.
0	CNT ZERO EVENT	Событие совпадения CNT с нулем 0 – нет события 1 – есть событие Сбрасывается записью 0, если запись одновременно с новым событием совпадения, приоритет у нового события. Если с момента совпадения до момента программного сброса CNT не изменил состояния, то флаг повторно не взводится.

1.7.13 IE

Таблица 316 – Регистр разрешения прерывания таймера IE

Номер	31...17	16...13	12...9	8...5
Доступ	U	R/W	R/W	R/W
Сброс	0	0	0	0
	-	CCR CAP1 EVENT IE [3:0]	CCR REF EVENT IE [3:0]	CCR CAP EVENT IE [3:0]

Номер	4	3	2	1	0
Доступ	R/W	R/W	R/W	R/W	R/W
Сброс	0	0	0	0	0
	BRK EVENT IE	ETR FE EVENT IE	ETR RE EVENT IE	CNT ARR EVENT IE	CNT ZERO EVENT IE

Таблица 317 – Описание бит регистра IE

№	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...17	-	Зарезервировано
16...13	CCR CAP1 EVENT IE [3:0]	Флаг разрешения прерывания по событию переднего фронта на выходе CAP1 каналов таймера 0 – нет прерывания 1 – прерывание разрешено Бит 0 – первый канал Бит 3 – четвертый канал
12...9	CCR REF EVENT IE[3:0]	Флаг разрешения прерывания по событию переднего фронта на выходе REF каналов таймера 0 – нет прерывания 1 – прерывание разрешено Бит 0 – первый канал

		Бит 3 – четвертый канал
--	--	-------------------------

ПРЕДВАРИТЕЛЬНО

8...5	CCR CAP EVENT IE [3:0]	Флаг разрешения прерывания по событию переднего фронта на выходе CAP каналов таймера 0 – нет прерывания 1 – прерывание разрешено Бит 0 – первый канал Бит 3 – четвертый канал
4	BRK EVENT IE	Флаг разрешения по триггерированному по PCLK состоянию входа BRK, 0 – нет прерывания 1 – прерывание разрешено
3	ETR FE EVENT IE	Флаг разрешения прерывания по заднему фронту на входе ETR 0 – нет прерывания 1 – прерывание разрешено
2	ETR RE EVENT IE	Флаг разрешения прерывания по переднему фронту на входе ETR 0 – нет прерывания 1 – прерывание разрешено
1	CNT ARR EVENT IE	Флаг разрешения прерывания по событию совпадения CNT и ARR 0 – нет прерывания 1 – прерывание разрешено
0	CNT ZERO EVENT IE	Флаг разрешения прерывания по событию совпадения CNT и нуля 0 – нет прерывания 1 – прерывание разрешено

1.7.14 DMA_RE

Таблица 318 – Регистр DMA_RE разрешения запросов DMA от прерываний таймера

Номер	31...17	16...13	12...9	8...5
Доступ	U	R/W	R/W	R/W
Сброс	0	0	0	0
	-	CCR CAP1 EVENT RE [3:0]	CCR REF EVENT RE [3:0]	CCR CAP EVENT RE [3:0]

Номер	4	3	2	1	0
Доступ	R/W	R/W	R/W	R/W	R/W
Сброс	0	0	0	0	0
	BRK EVENT RE	ETR FE EVENT RE	ETR RE EVENT RE	CNT ARR EVENT RE	CNT ZERO EVENT RE

Примечание – регистры DMA_RE1-DMA_RE4 влияют на настройки для индивидуального канала с 1 по 4 соответственно, а не для всех каналов как регистр DMA_RE.

Таблица 319 – Описание бит регистра DMA_RE

№	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...17	-	Зарезервировано
16...13	CCR CAP1 EVENT RE [3:0]	Флаг разрешения запроса DMA по событию переднего фронта на выходе CAP1 каналов таймера 0 – нет запроса DMA 1 – запрос DMA разрешен Бит 0 – первый канал Бит 3 – четвертый канал
12...9	CCR REF EVENT RE[3:0]	Флаг разрешения запроса DMA по событию переднего фронта на выходе REF каналов таймера 0 – нет запроса DMA 1 – запрос DMA разрешен Бит 0 – первый канал Бит 3 – четвертый канал
8...5	CCR CAP EVENT RE [3:0]	Флаг разрешения запроса DMA по событию переднего фронта на выходе CAP каналов таймера 0 – нет запроса DMA 1 – запрос DMA разрешен Бит 0 – первый канал Бит 3 – четвертый канал
4	BRK EVENT RE	Флаг разрешения по триггерированному по PCLK состоянию входа BRK, 0 – нет запроса DMA 1 – запрос DMA разрешен
3	ETR FE EVENT RE	Флаг разрешения запроса DMA по заднему фронту на входе ETR 0 – нет запроса DMA 1 – запрос DMA разрешен
2	ETR RE EVENT RE	Флаг разрешения запроса DMA по переднему фронту на входе ETR 0 – нет запроса DMA 1 – запрос DMA разрешен
1	CNT ARR EVENT RE	Флаг разрешения запроса DMA по событию совпадения CNT и ARR 0 – нет запроса DMA 1 – запрос DMA разрешен
0	CNT ZERO EVENT RE	Флаг разрешения запроса DMA по событию совпадения CNT и нуля 0 – нет запроса DMA 1 – запрос DMA разрешен

Контроллер АЦП

В микроконтроллере реализован 10-ти разрядный АЦП. С помощью АЦП можно оцифровать сигнал с 3 внешних аналоговых выводов и двух внутренних каналов, на которые выводится датчик температуры и источник опорного напряжения. Скорость выборки составляет до 500 тысяч преобразований в секунду.

Контроллер АЦП позволяет:

- оцифровать один из 3 внешних каналов;
- оцифровать значение встроенного датчика температуры;
- оцифровать значение встроенного источника опорного напряжения;
- оцифровать значение напряжения батарейного домена;
- осуществить автоматический опрос заданных каналов;
- выработать прерывание при выходе оцифрованного значения за заданные пределы.

Для осуществления преобразования требуется 28 тактов синхронизации CLK. В качестве синхросигнала может выступать частота процессора CPU_CLK либо частота ADC_CLK формируемая в блоке «Сигналы тактовой частоты». Выбор частоты осуществляется с помощью бита Cfg_REG_CLKS. В контроллере АЦП частота может быть поделена с помощью бит Cfg_REG_DIVCLK[3:0]. Максимальная частота CLK не может превышать 15 МГц.

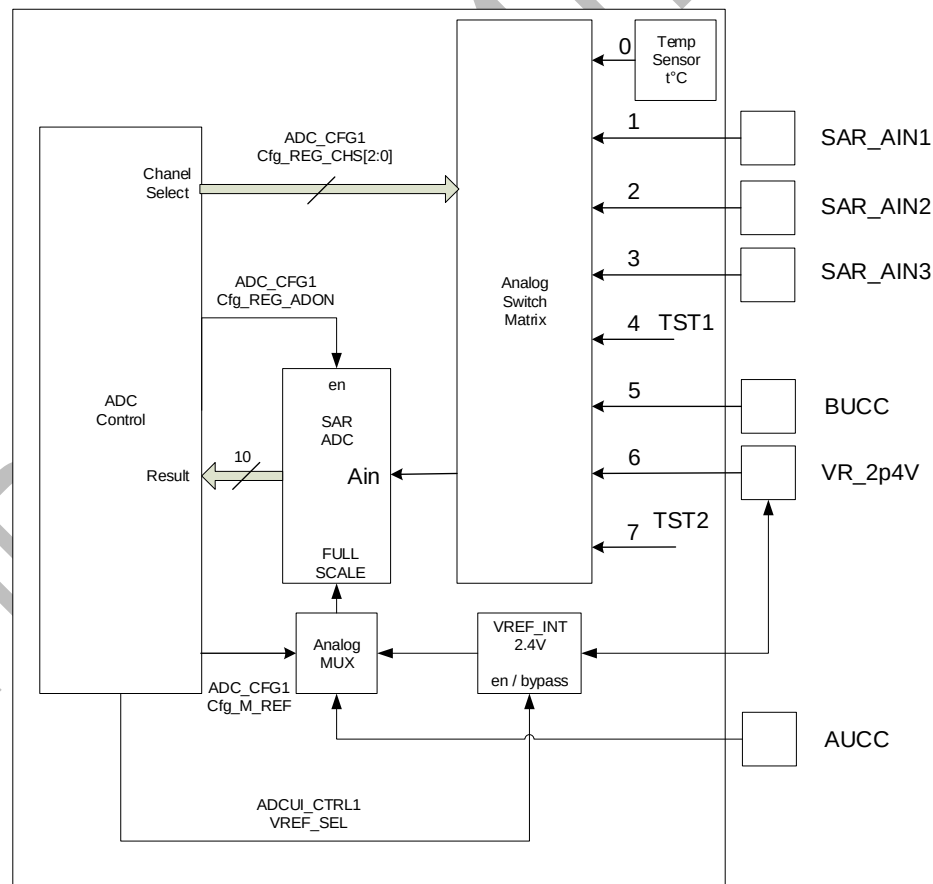


Рисунок 57. Структурная схема АЦП

Для включения АЦП необходимо установить бит Cfg_REG_ADON.

Преобразование внешнего канала

В регистре ADC1_CFG в битах Cfg_REG_CHS[2:0] необходимо задать соответствующий выводу номер канала. Преобразование может осуществляться при опоре равной AUCC (бит Cfg_M_REF = 0) и внешней (Cfg_M_REF = 1), в этом случае опора берется с вывода VR_2p4V. Биты Cfg_REG_CHCH, Cfg_REG_RNGC, Cfg_REG_SAMPLE должны быть сброшены.

Для начала преобразования необходимо записать 1 в бит Cfg_REG_GO. После завершения преобразования будет взведен бит Flg_REG_EOCIF в регистре ADC1_STATUS. А в регистре ADC1_RESULT будет результат преобразования. После считывания результата бит Flg_REG_EOCIF сбросится.

Если после первого преобразования результат не был считан и было выполнено второе преобразование, то в регистре результата ADCx_RESULT будет значение от последнего преобразования, а помимо бита Flg_REG_EOCIF будет взведен бит Flg_REG_OVERWRITE. Флаг Flg_REG_OVERWRITE может быть сброшен только записью в регистр ADC1_STATUS.

Последовательное преобразование нескольких каналов

Для автоматического последовательного преобразования нескольких каналов или одного канала в регистре ADC1_CHSEL необходимо установить единицы в битах, соответствующих необходимым для преобразования каналам. Преобразование может осуществляться при внутренней опоре (бит Cfg_M_REF = 0) и внешней (Cfg_M_REF = 1), в этом случае опора берется с выводов VR_2p4V. Бит Cfg_REG_RNGC должен быть сброшен, а Cfg_REG_SAMPLE и Cfg_REG_CHCH должны быть установлены. С помощью бит Delay_GO можно задать паузу между преобразованиями при переборе каналов. Для начала преобразования необходимо записать 1 в бит Cfg_REG_GO.

После завершения преобразования будет взведен бит Flg_REG_EOCIF в регистре ADC1_STATUS. А в регистре ADC1_RESULT будет результат преобразования. После считывания результата бит Flg_REG_EOCIF сбросится.

Если после первого преобразования результат не был считан и было выполнено второе преобразование, то в регистре результата ADC1_RESULT будет значение от последнего преобразования, а помимо бита Flg_REG_EOCIF будет взведен бит Flg_REG_OVERWRITE. Флаг Flg_REG_OVERWRITE может быть сброшен только записью в регистр ADC1_STATUS.

Для последовательного преобразования одного и того же канала можно в регистре ADC1_CHSEL выбрать только один канал и установить бит Cfg_REG_CHCH в 1, либо установить номер канала в битах Cfg_REG_CHS[2:0] и сбросить бит Cfg_REG_CHCH в 0. В этом случае процесс последовательного преобразования будет выполняться только для данного канала. Последовательное преобразование значения датчика температуры и источника опорного напряжения могут выполняться только в режиме последовательного преобразования одного канала.

Преобразование с контролем границ

При необходимости отслеживать нахождение оцифрованного значения в допустимых пределах можно задать нижнюю и верхнюю допустимые границы в регистрах ADC1_L_LEVEL и ADC1_H_LEVEL. При этом если установлен бит Cfg_REG_RNGC, то в случае если результат преобразования выходит за границы выставляется флаг Flg_REG_AWOIFEN. А в регистре результата будет полученное значение.

Формула расчета результата преобразования

напряжение на входе АЦП = (напряжение полной шкалы АЦП) * (значение регистра ADC1_RESULT/1024)

Датчик температуры

С помощью АЦП можно осуществить преобразования датчика температуры. Для выбора датчика температуры в качестве источника для преобразования необходимо в битах Cfg_REG_CHS установить значение 0 канала. После чего можно запустить процесс преобразования. Для начала преобразования необходимо записать 1 в бит Cfg_REG_GO.

После завершения преобразования будет взведен бит Flg_REG_EOCIF в регистре ADC1_STATUS. А в регистре ADC1_RESULT будет результат преобразования. После считывания результата бит Flg_REG_EOCIF сбросится.

Если после первого преобразования результат не был считан и было выполнено второе преобразование, то в регистре результата ADC1_RESULT будет значение от последнего преобразования, а помимо бита Flg_REG_EOCIF будет взведен бит Flg_REG_OVERWRITE. Флаг Flg_REG_OVERWRITE может быть сброшен только записью в регистр ADC1_STATUS.

Для последовательного преобразования только датчика температуры можно в регистре ADC1_CHSEL выбрать только 0 канал и установить бит Cfg_REG_CHCH в 1, либо установить номер 0-го канала в битах Cfg_REG_CHS[2:0] и сбросить бит Cfg_REG_CHCH в 0. В этом случае процесс последовательного преобразования будет выполняться только для данного канала.

Формула расчета температуры

Для определения значения температуры кристалла в градусах цельсия следует использовать формулу:

$$\text{TEMP} = 25 + ((\text{напряжение полной шкалы АЦП}) * (\text{значение регистра ADC1_RESULT}/1024) - 1.403)/0.0037$$

Например, при опросе 0 канала АЦП выдает код = 625, в качестве полной шкалы при этом выбран внешний источник 2.5В, подключенный ко входу микросхемы «VR_2p4V». Тогда температура кристалла равна:

$$\text{TEMP} = 25 + (2.5 * 625 / 1024 - 1.403) / 0.0037 = 58.2 \text{ C}$$

Время заряда внутренней емкости

Процесс преобразования состоит из двух этапов: сначала происходит заряд внутренней емкости до уровня внешнего сигнала, и затем происходит преобразование уровня заряда внутренней емкости в цифровой вид. Таким образом, для точного преобразования внешнего сигнала в цифровой вид, за время первого этапа внутренняя емкость должна зарядиться до уровня внешнего сигнала. Это время определяется соотношением номинальной внутренней емкости, входным сопротивлением тракта АЦП и выходным сопротивлением источника сигнала. Приведенная ниже формула позволяет определить максимальное выходное сопротивление источника R_{AIN} для обеспечения качественного преобразования:

$$R_{AIN} < \frac{T_{track}}{C_{ADC} \cdot \ln(2^N)} - R_{ADC} \quad (1)$$

$$T_{track} = 4 \cdot T_{ACLK} + N_{PCLKd} \cdot T_{PCLKd} = \frac{4}{f_{ACLK}} + \frac{(DelayGo+1)}{f_{PCLKd}} \quad (2)$$

где: T_{track} - время заряда внутренней емкости в тактах
 f_{ACLK} - рабочая частота АЦП
 C_{ADC} - внутренняя емкость АЦП (~2-3пФ)
 N - требуемая точность в разрядах
 R_{ADC} - входное сопротивление тракта АЦП (~1500 Ом)
 f_{PCLKd} – определяется формулой:

$$f_{PCLKd} = \frac{2^{Cfg_REG_DIVCLK}}{f_{PCLK}} \quad (3)$$

Если необходимо обеспечить преобразование с точностью 10 разрядов ± 1 LSB, то $N=10$. Время заряда T_s = определяется битами $DelayGo[2:0]$ и схемой самого АЦП и представлена в Таблица 320.

Время заряда внутренней емкости АЦП и время преобразования

Таблица 320 Время заряда внутренней емкости АЦП и время преобразования

DelayGo[2:0]	Дополнительная задержка перед началом преобразования	Общее время T_{track} заряда емкости АЦП перед началом преобразования	Общее время преобразования АЦП
000	1xPCLKd	4xACLK+1xPCLKd	28xACLK+1xPCLKd
001	2xPCLKd	4xACLK+2xPCLKd	28xACLK+2xPCLKd
010	3xPCLKd	4xACLK+3xPCLKd	28xACLK+3xPCLKd
011	4xPCLKd	4xACLK+4xPCLKd	28xACLK+4xPCLKd
100	5xPCLKd	4xACLK+5xPCLKd	28xACLK+5xPCLKd
101	6xPCLKd	4xACLK+6xPCLKd	28xACLK+6xPCLKd
110	7xPCLKd	4xACLK+7xPCLKd	28xACLK+7xPCLKd
111	8xPCLKd	4xACLK+8xPCLKd	28xACLK+8xPCLKd

Помимо точности определяемой временем зарядки внутренней емкости АЦП точность преобразования имеет ошибки, связанные с технологическими разбросами схемы и шумами и определяемые параметрами E_{DLADC} , E_{ILADC} и E_{OFFADC} .

Для корректного задания режимов работы АЦП в регистре $ADCx_CFG$ необходимо сделать до задания бита Cfg_REG_GO , иначе новая конфигурация будет действовать со следующего преобразования.

Описание регистров блока контроллера АЦП

Таблица 321 Описание регистров блока контроллера АЦП

Базовый Адрес	Название	Описание
0x4004_0000	ADC	Контроллер ADC
Смещение		
0x00	ADC1_CFG	Регистр управления ADC
0x08	ADC1_H_LEVEL	Регистр верхней границы ADC
0x10	ADC1_L_LEVEL	Регистр нижней границы ADC
0x18	ADC1_RESULT	Регистр результата ADC
0x20	ADC1_STATUS	Регистр статуса ADC
0x28	ADC1_CHSEL	Регистр выбора каналов перебора ADC
0x30	ADC_TRIM	Регистр настройки термодатчика

ADC1_CFG

Таблица 322 Регистр ADC1_CFG

Номер	31...28	27...25	24...20	19
Доступ	U	R/W	U	R/W
Сброс	0	0	0	0
	-	Delay Go [2:0]	-	en_noise_rng

Номер	18..16	15...12	11	10	9
Доступ	U	R/W	R/W	R/W	R/W
Сброс	0	0	0	0	0
	-	Cfg REG DIVCLK [3:0]	Cfg M_REF	Cfg REG RNGC	Cfg REG CHCH

Номер	8..7	6...4	3	2	1	0
Доступ	U	R/W	R/W	R/W	R/W	R/W
Сброс	0	0	0	0	0	0
	-	Cfg REG CHS [2:0]	Cfg REG SAMPLE	Cfg REG CLKS	Cfg REG GO	Cfg REG ADON

Таблица 323 Описание бит регистра ADC1_CFG

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...28	-	Зарезервировано
27...25	Delay Go [2:0]	Задержка перед началом следующего преобразования после завершения предыдущего при последовательном переборе каналов: 000 - 0 тактов CLK 001 - 1 такт CLK ... 111 - 7 тактов CLK
24...20	-	Должны быть в нуле для правильного функционирования
19	en_noise_rng	Бит включения rng генераторов для генерации шума по основному питанию микросхемы 1 - включено 0 - выключено
18..16	-	Должны быть в нуле для правильного функционирования
15...12	Cfg REG DIVCLK [3:0]	Выбор коэффициента деления частоты процессора: 0000 – PCLKd = HCLK; 0001 – PCLKd = HCLK/2; 0010 – PCLKd = HCLK/4; 0011 – PCLKd = HCLK/8; ... 1011 – PCLKd = HCLK/2048 Остальные PCLKd = HCLK
11	Cfg M_REF	Выбор шкалы АЦП: 0 – шкала АЦП от 0 до напряжения AUCC 1 – шкала АЦП от 0 до напряжения VR_2p4V
10	Cfg REG RNGC	Разрешение автоматического контролирования уровней: 1 – разрешено: выработка прерывания при выходе за диапазон в регистрах границы обработки; 0 – не разрешено
9	Cfg REG CHCH	Выбор переключения каналов: 1 – переключение включено (перебираются каналы, выбранные в регистре выбора канала Sl_Ch_Ch_REF); 0 – используется только выбранный канал
8..7	-	Должны быть в нуле для правильного функционирования
6...4	Cfg REG CHS [2:0]	Выбор аналогового канала, по которому поступает сигнал для преобразования: 111 – 0 канал (термодатчик) 110 – 1 канал (подключение к SAR_AIN1) 101 – 2 канал (подключение к SAR_AIN2) 100 – 3 канал (подключение к SAR_AIN3) 011 – 4 канал (тестовый режим) 010 – 5 канал (подключение к VDD_BD) 001 – 6 канал (подключение к VR_2p4V) 000 – 7 канал (тестовый режим)
3	Cfg REG	Выбор способа запуска АЦП: 1 – последовательный: автоматический запуск после

	SAMPLE	завершения предыдущего преобразования; 0 – одиночный
2	Cfg REG CLKS	Выбор источника синхросигнала CLK работы ADC: 1 – ACLK (формируется от выбранной частоты, подробнее см. раздел Сигналы тактовой частоты)); 0 – PCLK (формируется от частоты ядра, подробнее см. раздел Сигналы тактовой частоты).
1	Cfg REG GO	Начало преобразования Запись “1” начинает процесс преобразования, сбрасывается автоматически
0	Cfg REG ADON	Включение АЦП: 1 – включено; 0 – выключено

ADC1_H_LEVEL

Таблица 324 Регистр ADC1_H_LEVEL

Номер	31...10	9...0
Доступ	U	R/W
Сброс	0	0
	-	REG H LEVEL [9:0]

Таблица 325 Описание бит регистра ADC1_H_LEVEL

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...10	-	Зарезервировано
9...0	REG H LEVEL [9:0]	Верхняя граница зоны допуска.

ADC1_L_LEVEL

Таблица 326 Регистр ADC1_L_LEVEL

Номер	31...10	9...0
Доступ	U	R/W
Сброс	0	0
		REG L LEVEL [9:0]

Таблица 327 Описание бит регистра ADC1_L_LEVEL

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31..10	-	Зарезервировано
9...0	REG L LEVEL [9:0]	Нижняя граница зоны допуска.

ADC1_RESULT

Таблица 328 Регистр ADC1_RESULT

Номер	31...19	18...16	15...10	9...0
Доступ	U	RO	U	RO
Сброс	0	0	0	0
	-	CHANNEL [2:0]	-	RESULT [9:0]

Таблица 329 Описание бит регистра ADC1_RESULT

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...19	-	Зарезервировано
18...16	CHANNEL [2:0]	Канал результата преобразования: 111 – 0 канал (термодатчик) 110 – 1 канал (подключение к SAR_AIN1) 101 – 2 канал (подключение к SAR_AIN2) 100 – 3 канал (подключение к SAR_AIN3) 011 – 4 канал (тестовый режим) 010 – 5 канал (подключение к VDD_BD) 001 – 6 канал (подключение к VR_2p4V) 000 – 7 канал (тестовый режим)
15...10	-	Зарезервировано
9...0	RESULT [9:0]	Значение результата преобразования

ADC1_STATUS

Таблица 330 Регистр ADC1_STATUS

Номер	31...5	4	3	2	1	0
Доступ	U	R/W	R/W	R/W	R/W	R/W
Сброс	0	0	0	0	0	0
	-	ECOIF_IE	AWOIF_IE	Flg REG EOCIF	Flg REG AWOIFEN	Flg REG OVERWRITE

Таблица 331 Описание бит регистра ADC1_STATUS

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...5	-	Зарезервировано
4	ECOIF_IE	Флаг разрешения генерирования прерывания по событию Flg_REG_ECOIF: 0 – прерывание не генерируется; 1 – прерывание генерируется
3	AWOIF_IE	Флаг разрешения генерирования прерывания по событию Flg_REG_AWOIFEN: 0 – прерывание не генерируется; 1 – прерывание генерируется
2	Flg	Флаг выставляется, когда закончено преобразование и

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
	REG EOCIF	данные еще не считаны. Очищается считыванием результата из регистра ADCx_RESULT: 1 – есть готовый результат преобразования; 0 – нет результата
1	Flg REG AWOIFEN	Флаг выставляется, когда результат преобразования выше верхней или ниже нижней границы автоматического контроля уровней. Сбрасывается только при записи нуля в данный бит. 0 – результат в допустимой зоне 1 – вне допустимой зоны
0	Flg REG OVERWRITE	Данные в регистре результата были перезаписаны. Сбрасывается только при записи нуля в данный бит. 0 – не было события перезаписи не считанного результата; 1 – был результат преобразования, который не был считан

ADC1_CHSEL

Таблица 332 Регистр ADC1_CHSEL

Номер	31...8	7...0
Доступ	U	R/W
Сброс	0	0
	-	Sl_Ch_Ch_REF[7:0]

Таблица 333 Описание бит регистра ADC1_CHSEL

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31..8	-	Должны быть в нуле для правильного функционирования
7...0	Sl_Ch_Ch_REF[7:0]	Выбор каналов автоматического перебора: 0 – в соответствующем бите канал не участвует в переборе; 1 – в соответствующем бите канал участвует в переборе. Номера каналов, участвующих в переборе, соответствует номерам установленных бит в регистре.

ADC_TRIM

Таблица 334 Регистр ADC_TRIM

Номер	31..6	5	4	3	2	1	0
Доступ	U	R/W	R/W	R/W	R/W	R/W	U
Сброс	0	1	0	0	0	0	0
	-	BG_TRIM[4:0]					-

Таблица 335 Описание бит регистра ADCx_TRIM

Разряды	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...6	-	Зарезервировано
5...1	BG_TRIM[4:0]	Подстройка опорного напряжения bandgap. Рекомендуется не изменять значение бит установленных по умолчанию при сбросе.
0	-	Зарезервировано

Контроллер интерфейса I2C

I2C является двухпроводным, двунаправленным последовательным каналом связи с простым и эффективным методом обмена данными между устройствами. Интерфейс применяется, когда надо организовать обмен на коротком расстоянии между несколькими устройствами. Стандарт интерфейса I2C является многомастерным с обнаружением коллизий и арбитражем, исключающим потерю данных при обмене, когда два или более мастера пытаются осуществить передачу одновременно. Контроллер интерфейса I2C в микроконтроллере работает только в режиме Master.

Интерфейс работает на 3 скоростях:

- нормальная: 100 Kbps;
- быстрая: 400 Kbps;
- очень быстрая: 1 Mbps.

Скорость обмена данными блоком I2C рассчитывается по формуле:

$$F_{scl} = HCLK / (5 * (DIV + 1)).$$

Конфигурация системы

I2C системы используют последовательную линию данных SDA и линию тактового сигнала SCL. Все устройства, подсоединенные к этим двум линиям, должны работать в режиме открытого стока, обеспечивая тем самым создание на линии «проводного И» за счет внешних резисторов подтяжки обеих линий к питанию.

Передача данных между мастером и ведомым осуществляется по линии SDA и синхронизируется по линии SCL. После завершения передачи информации осуществляется передача в обратную сторону одного бита подтверждения. Каждый принимаемый бит фиксируется принимающей стороной при высоком уровне SCL и может изменяться передатчиком при низком уровне. Изменение линии SDA при высоком уровне SCL является командным состоянием (см. «Сигнал START» и «Сигнал STOP»).

Протокол I2C

Нормальная передача по интерфейсу I2C содержит 4 фазы:

- сигнал START;
- передача адреса;
- передача данных;
- сигнал STOP.

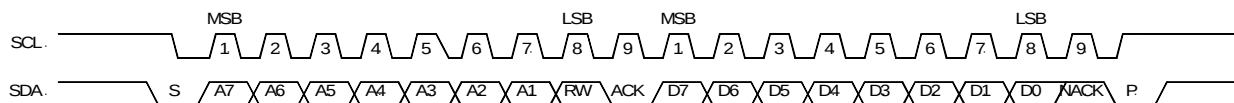


Рисунок 58. Передача по I2C

Сигнал START

Когда шина находится в свободном состоянии, т.е. не одно из устройств не осуществляет передачи (на линиях SCL и SDA высокий уровень), мастер может инициализировать процесс передачи через создание сигнала START на линии. Сигнал START или S бит задается, когда уровень на линии SDA переходит из высокого в низкий при высоком уровне на линии SCL. Появление сигнала START не означает начала передачи данных.

Повторный сигнал START является обычным сигналом START, но без предварительно сгенерированного до этого сигнала STOP. Мастер может использовать метод для начала соединения с другим ведомым или с тем же ведомым, но с изменением режима работы (например, чтение после записи, или, наоборот) без перевода шины в свободное состояние.

Контроллер интерфейса генерирует сигнал START при записи единицы в бит START регистра I2C_CMD при установленных битах RD или WR. В зависимости от состояния линии SCL генерируется либо сигнал START, либо повторный сигнал START.

Передача адреса

Первым байтом данных, передаваемым мастером сразу после сигнала START, является адрес ведомого. Это 7-ми битный адрес и следующий за ним бит RW. Бит RW определяет дальнейшее направление передачи данных. В системе на одной шине не может быть несколько ведомых устройств с одним адресом. Ведомое устройство, у которого совпадает адрес с адресом в сообщении, подтверждает прием, выставя ACK и опуская линию SDA в низкий уровень на 9-й SCL тактовый импульс. Контроллер также поддерживает 10-битный адрес путем генерации двух циклов передачи адреса.

Процесс выдачи адреса выполняется как цикл записи. Необходимо записать адрес ведомого в регистр I2C_TXD и установить бит WR в регистре I2C_CMD. Контроллер осуществит передачу адреса в линию.

Передача данных

После успешного подтверждения приема адреса одним ведомым устройством может быть начата передача данных в направлении, задаваемым битом RW в посылке мастера. Каждый передаваемый бит подтверждается ACK на 9-й SCL тактовый импульс. Если ведомое устройство выдало NACK (нет подтверждения), то мастер может сгенерировать либо сигнал STOP для прекращения передачи, либо повторный сигнал START для начала нового цикла передачи.

Если мастер является принимающим устройством и выдает NACK, то ведомое устройство отпускает линию SDA и мастер может сгенерировать сигнал STOP или повторный сигнал START.

Для записи данных в ведомое устройство запишите данные в регистр I2C_TXD и установите бит WR. Для чтения данных из устройства установите бит RD. На время выполнения передачи контроллер интерфейса выставляет флаг TR_PROG в регистре I2C_STA. Когда передача завершена, этот флаг снимается и устанавливается флаг INT. Если при этом установлен бит разрешения INT_EN, то генерируется прерывание контроллеру прерываний. Регистр I2C_RXD содержит корректные принятые данные после установки флага INT. Пользователь может начать новый цикл чтения или записи только тогда, когда флаг TR_PROG сброшен.

Сигнал STOP

Мастер может завершить соединение путем создания сигнала STOP. Сигнал STOP или Р бит определяется переходом линии SDA из низкого состояния в высокое, когда SCL находится в высоком состоянии.

Описание регистров контроллера I2C

Таблица 336 – Описание регистров контроллера I2C

Базовый Адрес	Название	Описание
0x4003_0000	I2C	Контроллер I2C
Смещение		
0x00	PRL	Младшая часть пред делителя частоты
0x04	PRH	Старшая часть пред делителя частоты
0x08	CTR	Управление контроллером I2C
0x0C	RXD	Принятые данные по I2C
0x10	STA	Статус I2C
0x14	TXD	Передаваемые данные по I2C
0x18	CMD	Управление I2C

PRL

Таблица 337 – Регистр PRL

Номер	31...8	7... 0
Доступ	U	R/W
Сброс	0	0
	-	PR[7:0]

Таблица 338 – Описание бит регистра PRL

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...8	-	Зарезервировано
7...0	PR[7:0]	Младшая часть предделителя

PRH

Таблица 339 – Регистр PRH

Номер	31...8	7... 0
Доступ	U	R/W
Сброс	0	0
	-	PR[15:8]

Таблица 340 – Описание бит регистра PRH

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...8	-	Зарезервировано

7...0	PR[15:8]	Старшая часть делителя
-------	----------	------------------------

CTR

Таблица 341 – Регистр CTR

Номер	31...8	7	6	5	4...0
Доступ	U	R/W	R/W	R/W	U
Сброс	0	0	0	0	0
	-	EN_I2C	EN_INT	S_I2C	-

Таблица 342 – Описание бит регистра CTR

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...8	-	Зарезервировано
7	EN_I2C	Разрешение работы контроллера I2C: 0 – выключен; 1 – включен
6	EN_INT	Разрешение прерывания от I2C: 0 – запрещено; 1 – разрешено
5	S_I2C	Скорость работы I2C: 0 – до 400 кГц; 1 – до 1 МГц
4...0	-	Зарезервировано

RXD

Таблица 343 – Регистр RXD

Номер	31...8	7... 0
Доступ	U	R/W
Сброс	0	0
	-	RXD[7:0]

Таблица 344 – Описание бит регистра RXD

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...8	-	Зарезервировано
7...0	RXD[7:0]	Последний полученный по I2C байт

STA

Таблица 345 – Регистр STA

Номер	31...8	7	6	5	4...2	1	0
Доступ	U	R/W	R/W	R/W	U	R/W	R/W
Сброс	0	0	0	0	0	0	0
	-	Rx ACK	BUSY	LOST ARB	-	TR PROG	INT

Таблица 346 – Описание бит регистра STA

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...8	-	Зарезервировано
7	Rx ACK	Полученный от ведомого ACK: 0 – ACK получен; 1 – получен NACK
6	BUSY	Состояние шины I2C: 0 – после получения Stop bit; 1 – после получения состояния Start bit
5	LOST ARB	Потеря арбитража: 0 – нет потери арбитража; 1 – потерян арбитраж. Этот бит выставляется если: - получен Stop bit, но он не был инициализирован этим контроллером; - Если контроллер пытается выставить SDA в высокий уровень, но SDA остается в низком
4...2	-	Зарезервировано
1	TR PROG	Процесс передачи: 0 – передача завершена; 1 – передаются данные
0	INT	Флаг прерывания, выставляется всегда. Прерывание для процессора выдается, если есть флаг EN_INT: 0 – нет прерывания; 1 – есть прерывание. Флаг выставляется если: - передача байта завершена; - был потерян арбитраж

TXD

Таблица 347 – Регистр TXD

Номер	31...8	7... 0
Доступ	U	R/W
Сброс	0	0
	-	TXD[7:0]

Таблица 348 – Описание бит регистра TXD

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...8	-	Зарезервировано
7...0	TXD[7:0]	Байт для отправки по I2C. При передаче адреса нулевой бит определяет режим передачи: 0 – запись в ведомое устройство; 1 – чтение из ведомого устройства

CMD

Таблица 349 – Регистр CMD

Номер	31...8	7	6	5	4	3	2...1	0
Доступ	U	R/W	R/W	R/W	R/W	R/W	U	R/W
Сброс	0	0	0	0	0	0	0	0
	-	START	STOP	RD	WR	ACK	-	CLR INT

Таблица 350 – Описание бит регистра CMD

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...8	-	Зарезервировано
7	START	Отправить START bit. Инициализируется записью 1. После завершения отправки автоматически не сбрасывается, очищается записью нуля
6	STOP	Отправить STOP bit. Инициализируется записью 1. После завершения отправки автоматически не сбрасывается, а очищается записью нуля
5	RD	Чтение из ведомого: 0 – нет действия; 1 – начать чтение
4	WR	Запись в ведомого; 0 – нет действия; 1 – начать запись
3	ACK	Отправить ACK: 0 – отправить ACK; 1 – отправить NACK
2...1	-	Зарезервировано
0	CLR	Очистить прерывание INT. Запись 1 очищает прерывание

	INT	
--	-----	--

ПРЕДВАРИТЕЛЬНО

Контроллер SSP

Модуль порта синхронной последовательной связи (SSP – Synchronous Serial Port) выполняет функции интерфейса последовательной синхронной связи в режиме ведущего и ведомого устройства и обеспечивает обмен данными с подключенным ведомым или ведущим периферийным устройством в соответствии с одним из протоколов:

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

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

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

Модуль формирует сигналы прерываний по следующим событиям:

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

Основные сведения о модуле представлены в следующих разделах:

- характеристики интерфейса SPI;
- характеристики интерфейса Microwire;
- характеристики интерфейса SSI.

1.1 Основные характеристики модуля SSP

- Функционирует как в ведущем, так и в ведомом режиме;
- Программное управление скоростью обмена;
- Состоит из независимых буферов приема и передачи (8 ячеек по 16 бит) с организацией доступа типа FIFO (First In First Out – первый вошел, первый вышел);
- Программный выбор одного из интерфейсов обмена: SPI, Microwire, SSI;
- Программируемая длительность информационного кадра от 4 до 16 бит;
- Независимое маскирование прерываний от буфера FIFO передатчика, буфера FIFO приемника, а также по переполнению буфера приемника;
- Доступна возможность тестирования по шлейфу, соединяющему вход с выходом;
- Поддержка прямого доступа к памяти (DMA).

Структурная схема модуля представлена на рисунке 59.

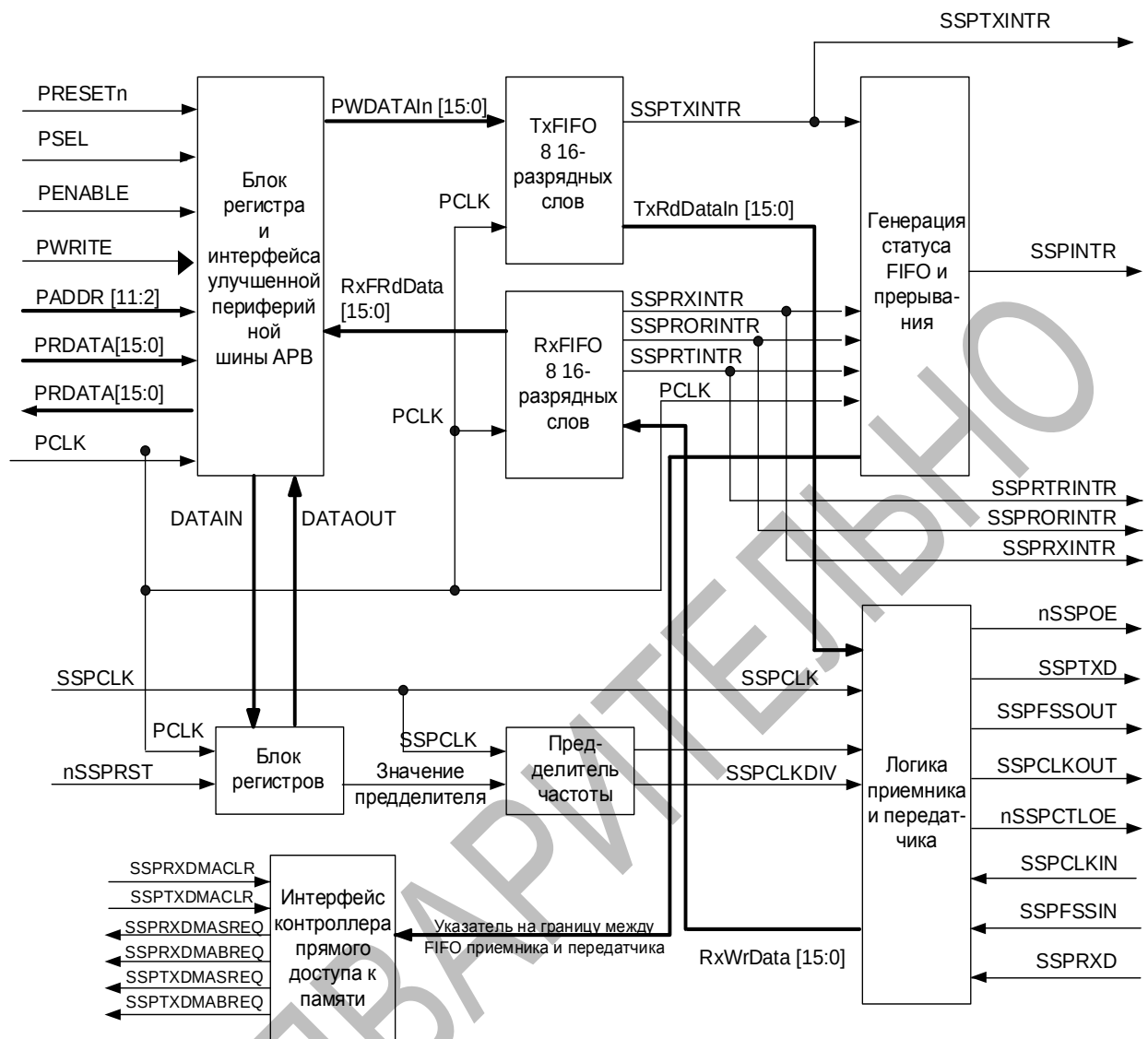


Рисунок 59 – Структурная схема модуля SSP

1.2 Программируемые параметры

Следующие ключевые параметры могут быть заданы программно:

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

1.3 Характеристики интерфейса SPI

Последовательный синхронный интерфейс SPI фирмы Motorola обеспечивает:

- полнодуплексный обмен данными по четырехпроводной линии;
- программное задание фазы и полярности тактового сигнала.

1.4 Характеристики интерфейса Microwire

Интерфейс Microwire фирмы National Semiconductor обеспечивает:

- полудуплексный обмен данными с использованием 8-битных управляющих последовательностей.

1.5 Характеристики интерфейса SSI

Интерфейс SSI фирмы Texas Instruments обеспечивает:

- полнодуплексный обмен данными по четырехпроводной линии;
- возможность перевода линии передачи данных в третье (высокоимпедансное) состояние.

1.6 Общий обзор модуля SSP

Модуль SSP представляет собой интерфейс синхронного последовательного обмена данными способный функционировать в качестве ведущего или ведомого устройства, поддерживающий протоколы передачи данных SPI фирмы Motorola, Microwire фирмы National Semiconductor, а также SSI фирмы Texas Instruments.

Модуль выполняет следующие функции:

- преобразование данных, полученных от периферийного устройства, из последовательной в параллельную форму;
- преобразование данных, передаваемых на периферийное устройство, из параллельной в последовательную форму;
- центральный процессор читает и записывает данные, а также управляющую информацию и информацию о состоянии;
- прием и передача данных буферизуются с помощью буферов FIFO и обеспечивают хранение до восьми слов данных шириной до 16 бит независимо для режимов приема и передачи.

Последовательные данные передаются по линии SSPTXD и принимаются с линии SSPRXD.

Модуль SSP содержит программируемые делители частоты, формирующие тактовый сигнал обмена данными SSPCLKOUT (сигнал, поступающий на линию SSPSCKOUT и предназначенный для синхронизации приема и передачи данных) из сигнала, поступающего на линию SSPCLK (сигнал, который подается на модуль SSP с блока формирования тактовых частот). Скорость передачи данных может быть более 2 МГц, в зависимости от частоты SSPCLK и характеристик подключенного периферийного устройства.

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

Модуль формирует четыре независимо маскируемых прерывания:

- | | |
|------------|---|
| SSPTXINTR | – запрос на обслуживание буфера передатчика; |
| SSPRXINTR | – запрос на обслуживание буфера приемника; |
| SSPRORINTR | – переполнение приемного буфера FIFO; |
| SSPRTINTR | – таймаут ожидания чтения данных из приемного FIFO. |

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

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

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

1.6.1 Блок формирования тактового сигнала

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

Коэффициент предварительного деления частоты в диапазоне от 2 до 254 с шагом 2 можно задать путем записи значения в регистр CPSR. Так как младший значащий разряд коэффициента деления не используется, то исключается возможность деления частоты на нечетный коэффициент деления. Это, в свою очередь, гарантирует формирование тактового сигнала симметричной формы (с одинаковой длительностью полупериодов высокого и низкого уровней).

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

Коэффициент деления второго делителя задается программно в диапазоне от 1 до 256 путем записи соответствующего значения в регистр управления CR0.

1.6.2 Буфер FIFO передатчика

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

В случае работы контроллера SPI в режиме SLAVE, чтение пустого FIFO приводит к выдаче некорректных данных.

1.6.3 Буфер FIFO приемника

Буфер приемника имеет ширину 16 бит, глубину 8 слов, схему организации доступа типа FIFO («первый вошел, первый вышел»). Принятые от периферийного устройства данные сохраняются в этом буфере блоком приема данных до тех пор, пока не будут считаны центральным процессором.

1.6.4 Блок приема и передачи данных

Режим ведущего устройства

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

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

Блок приемника выполняет преобразование данных, поступающих синхронно с линии SSPRXD, из последовательной в параллельную форму. После этого загружает их в буфер FIFO приемника, откуда они могут быть считаны процессором.

Режим ведомого устройства

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

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

Аналогично, блок приемника выполняет преобразование данных, поступающих с линии SSPRXD синхронно с сигналом SSPCLKIN, из последовательной в параллельную форму, после чего загружает их в буфер FIFO приемника, откуда они могут быть считаны процессором.

1.6.5 Блок формирования прерываний

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

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

1.6.6 Интерфейс прямого доступа к памяти

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

1.6.7 Конфигурирование приемопередатчика

После сброса работа блоков приемопередатчика запрещается до выполнения процедуры задания конфигурации.

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

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

1.6.8 Разрешение работы приемопередатчика

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

После разрешения работы модуля приемопередатчик начинает обмен данными по линиям SSPTXD и SSPRXD.

1.6.9 Соотношения между тактовыми сигналами

В модуле имеется ограничение на соотношение между частотами тактовых сигналов CPU_CLK и SSPCLK. Частота SSPCLK должна быть меньше или равна частоте CPU_CLK. Выполнение этого требования гарантирует синхронизацию сигналов управления, передаваемых из зоны действия тактового сигнала SSPCLK в зону действия сигнала CPU_CLK в течение времени, меньшего продолжительности передачи одного информационного кадра:

$$F_{SSPCLK} \leq F_{PCLK}$$

В режиме ведомого устройства сигнал SSPCLKIN от ведущего внешнего устройства поступает на схемы синхронизации, задержки и обнаружения фронта. Для того чтобы обнаружить фронт сигнала SSPCLKIN, необходимо три такта сигнала SSPCLK. Сигнал SSPTXD имеет меньшее время установки по отношению к заднему фронту SSPCLKIN, по которому и происходит считывание данных из линии. Время установки и удержания сигнала SSPRXD по отношению к сигналу SSPCLKIN должно выбираться с запасом, гарантирующим правильное считывание данных. Для обеспечения корректной работы устройства необходимо, чтобы частота SSPCLK была как минимум в 12 раз больше, чем максимальная предполагаемая частота сигнала SSPCLKIN.

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

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

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

Соотношение между максимальной частотой сигнала SSPCLK и минимальной частотой SSPCLKOUT составляет 254×256 .

Минимальная допустимая частота сигнала SSPCLK определяется следующей системой соотношений, которые должны выполняться одновременно:

$$\begin{aligned} F_{\text{SSPCLK}}(\text{min}) &=> 2 \times F_{\text{SSPCLKOUT}}(\text{max}) \text{ [for master mode]} \\ F_{\text{SSPCLK}}(\text{min}) &=> 12 \times F_{\text{SSPCLKIN}}(\text{max}) \text{ [for slave mode]}. \end{aligned}$$

Аналогично, максимальная допустимая частота сигнала SSPCLK определяется следующей системой соотношений, которые должны выполняться одновременно:

$$\begin{aligned} F_{\text{SSPCLK}}(\text{max}) &\leq 254 \times 256 \times F_{\text{SSPCLKOUT}}(\text{min}) \text{ [for master mode]} \\ F_{\text{SSPCLK}}(\text{max}) &\leq 254 \times 256 \times F_{\text{SSPCLKIN}}(\text{min}) \text{ [for slave mode]}. \end{aligned}$$

1.6.10 Программирование регистра управления CR0

Регистр CR0 предназначен для:

- установки скорости информационного обмена;
- выбора одного из трех протоколов обмена данными;
- выбора размера слова данных.

Скорость информационного обмена зависит от частоты внешнего тактового сигнала SSPCLK и коэффициента деления блока формирования тактового сигнала. Последний задается совместно значением поля SCR (Serial Clock Rate – скорость информационного обмена) регистра CR0 и значением поля CPSDVSР (clock prescale divisor value – коэффициент деления тактового сигнала) регистра CPSR.

Формат информационного кадра задается путем установки значения поля FRF, а размер слова данных – путем установки значения поля DSS регистра CR0.

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

1.6.11 Программирование регистра управления CR1

Регистр CR1 предназначен для:

- выбора ведущего или ведомого режима функционирования приемопередатчика;
- включения режима проверки канала по шлейфу;
- разрешения или запрещения работы модуля.

Выбор ведущего режима осуществляется путем записи 0 в поле MS регистра CR1 (это значение устанавливается после сброса автоматически).

Запись 1 в поле MS переводит приемопередатчик в режим ведомого устройства. В этом режиме разрешение или запрещение формирования сигнала передатчика SSPTXD осуществляется путем установки бита SOD (slave mode SSPTXD output disable – запрет линии SSPTXD для ведомого режима) регистра CR1. Указанная функция полезна при подключении к одной линии нескольких подчиненных устройств.

Для того чтобы разрешить функционирование приемопередатчика, необходимо установить в 1 бит SSE (Synchronous Serial Port Enable – разрешение последовательного синхронного порта).

1.6.12 Формирование тактового сигнала обмена данными

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

Частота выходного тактового сигнала обмена данными SSPCLK определяется следующим соотношением:

$$F_{SSPCLK} = F_{SSPCLK} / (CPSDVSR * (1+SCR)).$$

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

1.6.13 Формат информационного кадра

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

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

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

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

В режиме построения кадра SSI фирмы Texas Instruments перед началом каждого информационного кадра на выходе SSPFSS формируется импульс с длительностью, равной одному тактовому интервалу обмена данными. В этом режиме приемопередатчик SSP, равно как и ведомые периферийные устройства, передает данные в линию по переднему фронту сигнала SSPCLK, а считывает данные из линии по заднему фронту этого сигнала.

В отличие от полнодуплексных режимов передачи данных SSI и SPI, режим Microwire фирмы National Semiconductor использует специальный способ обмена данными между ведущим и ведомым устройством, функционирующий в режиме полудуплекса. В указанном режиме на внешнее ведомое устройство перед началом передачи информационного кадра посылается специальная восьмибитная управляющая последовательность. В течение всего времени передачи этой последовательности приемник не обрабатывает каких-либо входных данных. После того как сигнал передан и декодирован ведомым устройством, оно выдерживает паузу в один тактовый интервал после передачи последнего бита управляющей последовательности, после чего передает в адрес ведущего устройства запрошенные данные. Длительность блока данных от ведомого устройства может составлять от 4 до 16 бит, таким образом общая длительность информационного кадра составляет от 13 до 25 бит.

1.6.14 Формат синхронного обмена SSI фирмы Texas Instruments

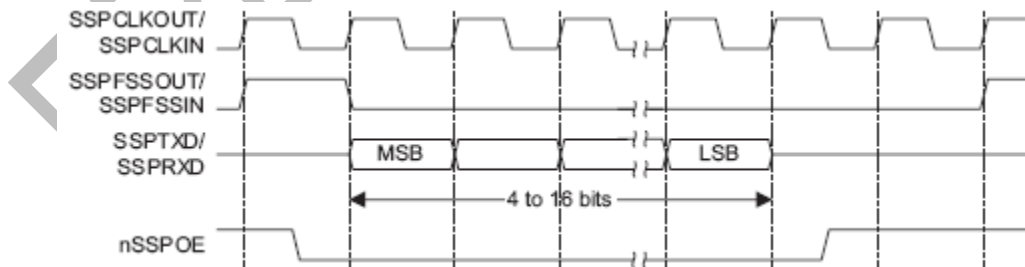


Рисунок 60 – Формат синхронного обмена протокола SSI (единичный обмен)

В данном режиме при неактивном приемопередатчике SSP сигналы SSPCLK и SSPFSS переводятся в низкий логический уровень, а линия передачи данных SSPTXD поддерживается в третьем состоянии.

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

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

Временные диаграммы последовательного синхронного обмена по протоколу SSI фирмы Texas Instruments показывает Рисунок 60 (передача единичного информационно кадра) и Рисунок 61 (передача последовательности кадров).

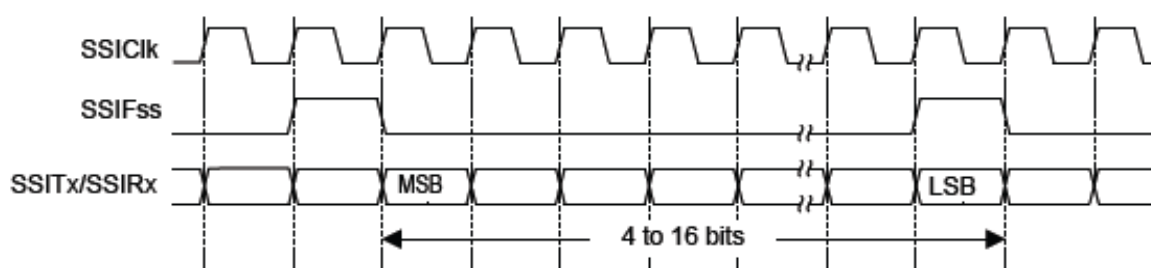


Рисунок 61 – Формат синхронного обмена протокола SSI (непрерывный обмен)

1.6.15 Формат синхронного обмена SPI фирмы Motorola

Интерфейс SPI фирмы Motorola осуществляется по четырем сигнальным линиям, при этом сигнал SSPFSS выполняет функцию выбора ведомого устройства. Главной особенностью протокола SPI является возможность выбора состояния и фазы сигнала SSPCLK в режиме ожидания (неактивном приемопередатчике) путем задания значений бит SPO и SPH регистра управления CR0.

Выбор полярности тактового сигнала – бит SPO

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

Выбор фазы тактового сигнала – бит SPH.

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

В случае если бит SPH установлен в 0, регистрация данных приемником осуществляется по нарастающему фронту тактового сигнала, в противном случае по спадающему фронту.

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

Временные диаграммы последовательного синхронного обмена в режиме SPI с SPO=0, SPH=0 показывает Рисунок 62 (одиночный обмен) и Рисунок 63 (непрерывный обмен).

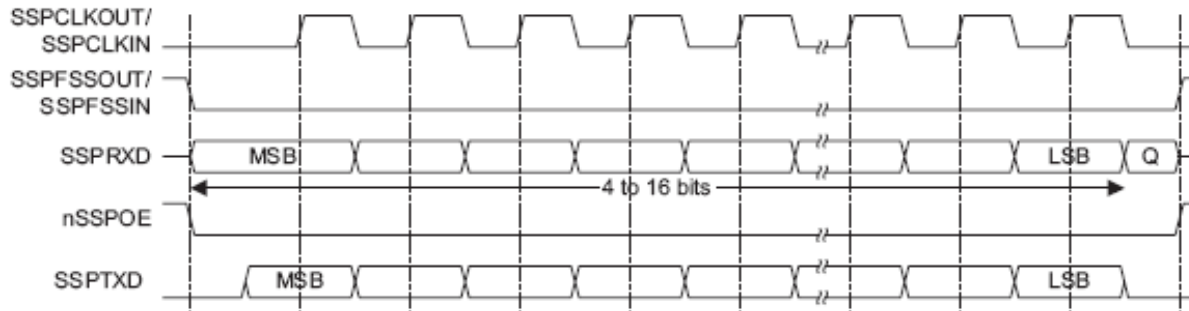


Рисунок 62 – Формат синхронного обмена протокола SPI, SPO=0, SPH=0 (одиночный обмен)

Примечание – На рисунке буквой Q обозначен сигнал с неопределенным уровнем.

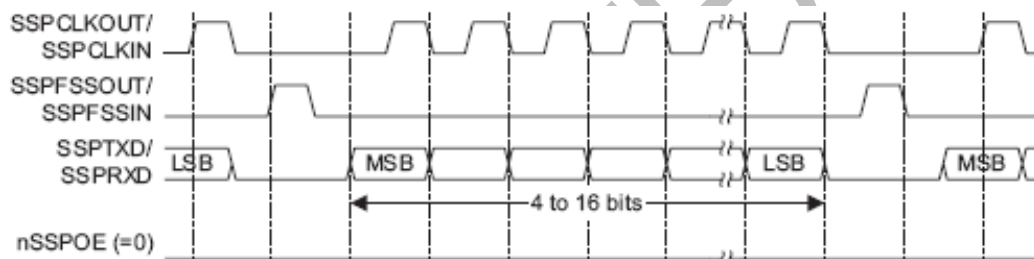


Рисунок 63 – Формат синхронного обмена протокола SPI, SPO=0, SPH=0 (непрерывный обмен)

В данном режиме во время ожидания приемопередатчика:

- сигнал SSPCLK имеет низкий логический уровень;
- сигнал SSPFSS имеет высокий логический уровень;
- сигнал SSPTXD переводится в высокоимпедансное состояние.

Если работа модуля разрешена и в буфере FIFO передатчика содержатся корректные данные, сигнал SSPFSS переводится в низкий логический уровень, что указывает на начало обмена данными и разрешает передачу данных от ведомого устройства на входную линию SSPRXD ведущего. При этом линия передатчика SSPTXD переходит из высокоимпедансного в активное состояние.

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

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

В случае передачи одного слова данных после приема его последнего бита линия SSPFSS переводится в высокий логический уровень по истечении одного периода тактового сигнала SSPCLK.

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

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

Временные диаграммы последовательного синхронного обмена в режиме SPI с SPO=0, SPH=1 показывает Рисунок 64 (одиночный и непрерывный обмен).

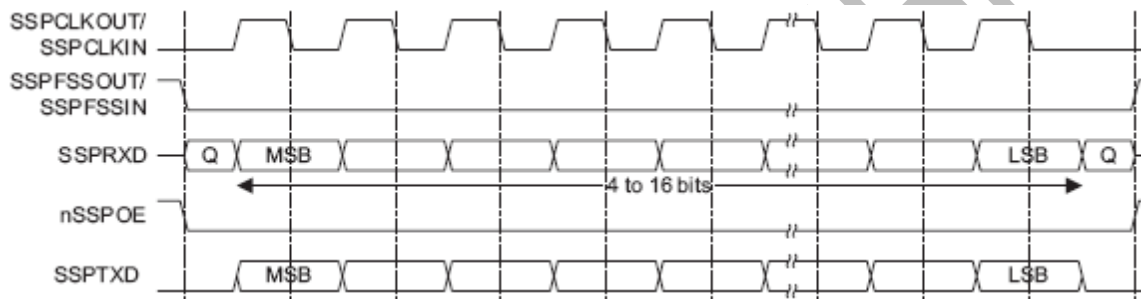


Рисунок 64 – Формат синхронного обмена протокола SPI, SPO=0, SPH=1

Примечание – На рисунке буквой Q обозначен сигнал с неопределенным уровнем.

В данном режиме во время ожидания приемопередатчика:

- сигнал SSPCLK имеет низкий логический уровень;
- сигнал SSPFSS имеет высокий логический уровень;
- сигнал SSPTXD переводится в высокоимпедансное состояние.

Если работа модуля разрешена и в буфере FIFO передатчика содержатся корректные данные, сигнал SSPFSS переводится в низкий логический уровень, что указывает на начало обмена данными и разрешает передачу данных от ведомого устройства на входную линию SSPRXD ведущего. Выходной контакт передатчика SSPTXD переходит из высокоимпедансного в активное состояние.

По истечении полутакта сигнала SSPCLK на линиях обмена, как ведущего, так и ведомого устройств будут сформированы значения первых бит передаваемых данных. В это же время включается линия SSPCLK и на ней формируется передний фронт сигнала.

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

В случае передачи одного слова данных после приема его последнего бита линия SSPFSS переводится в высокий логический уровень по истечении одного периода тактового сигнала SSPCLK.

В режиме непрерывной передачи данных линия SSPFSS постоянно находится в низком логическом уровне, и переводится в высокий уровень по окончании приема последнего бита блока данных, как и в режиме передачи одного слова.

ПРЕДВАРИТЕЛЬНО

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

Временные диаграммы последовательного синхронного обмена в режиме SPI с SPO=1, SPH=0 показывает Рисунок 65 (одиначный обмен) и Рисунок 66 (непрерывный обмен).

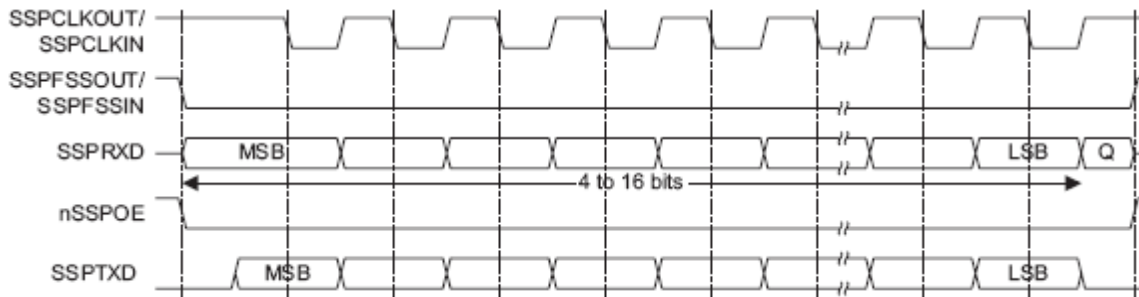


Рисунок 65 – Формат синхронного обмена протокола SPI, SPO=1, SPH=0 (одиначный обмен)

Примечание – На рисунке буквой Q обозначен сигнал с неопределенным уровнем.

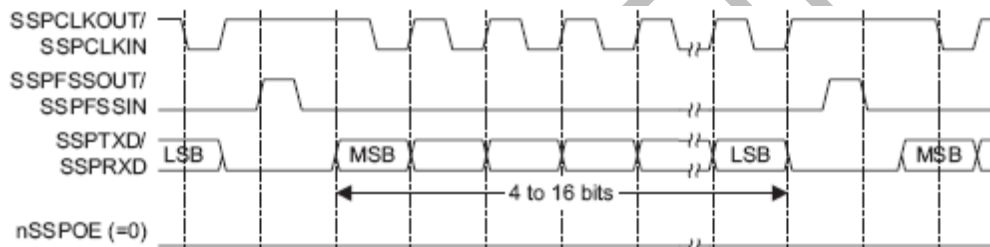


Рисунок 66 – Формат синхронного обмена протокола SPI, SPO=1, SPH=0 (непрерывный обмен)

В данном режиме во время ожидания приемопередатчика:

- сигнал SSPCLK имеет высокий логический уровень;
- сигнал SSPFSS имеет высокий логический уровень;
- сигнал SSPTXD переводится в высокоимпедансное состояние.

Если работа модуля разрешена и в буфере FIFO передатчика содержатся корректные данные, сигнал SSPFSS переводится в низкий логический уровень, что указывает на начало обмена данными и разрешает передачу данных от ведомого устройства на входную линию SSPRXD ведущего. Выходной контакт передатчика SSPTXD переходит из высокоимпедансного в активное состояние.

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

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

В случае передачи одного слова данных после приема его последнего бита линия SSPFSS переводится в высокий логический уровень по истечении одного периода тактового сигнала SSPCLK.

В режиме непрерывной передачи данных на линии SSPFSS должны формироваться импульсы высокого логического уровня между передачами каждого из слов данных. Это

связано с тем, что в режиме SPH=0 линия выбора ведомого устройства в низком уровне блокирует запись в сдвиговый регистр. Поэтому ведущее устройство должно переводить линию SSPFSS в высокий уровень по окончании передачи каждого кадра, разрешая запись новых данных. По окончании приема последнего бита блока данных линия SSPFSS переводится в состояние, соответствующее режиму ожидания, по истечении одного такта сигнала SSPCLK.

1.6.19 Формат синхронного обмена SPI фирмы Motorola, SPO=1, SPH=1

Временные диаграммы последовательного синхронного обмена в режиме SPI с SPO=1, SPH=1 показывает Рисунок 67 (одиночный и непрерывный обмен).

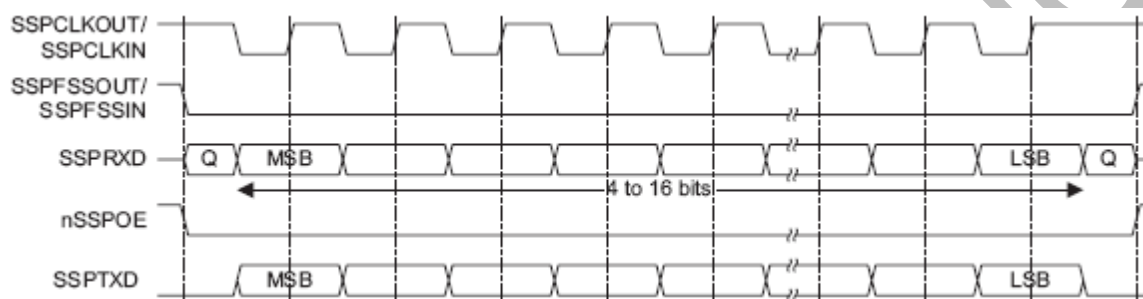


Рисунок 67 – Формат синхронного обмена протокола SPI, SPO=1, SPH=1

Примечание – На рисунке буквой Q обозначен сигнал с неопределенным уровнем.

В данном режиме во время ожидания приемопередатчика:

- сигнал SSPCLK имеет высокий логический уровень;
- сигнал SSPFSS имеет высокий логический уровень;
- сигнал SSPTXD переводится в высокоимпедансное состояние.

Если работа модуля разрешена и в буфере FIFO передатчика содержатся корректные данные, сигнал SSPFSS переводится в низкий логический уровень, что указывает на начало обмена данными и разрешает передачу данных от ведомого устройства на входную линию SSPRXD ведущего. Выходной контакт передатчика SSPTXD переходит из высокоимпедансного в активное состояние.

По истечении полутака сигнала SSPCLK на линиях обмена, как ведущего, так и ведомого устройств сформированы значения первых бит передаваемых данных. В это же время включается линия SSPCLK и на ней формируется передний фронт сигнала.

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

В случае передачи одного слова данных после приема его последнего бита линия SSPFSS переводится в высокий логический уровень по истечении одного периода тактового сигнала SSPCLK.

В режиме непрерывной передачи данных линия SSPFSS постоянно находится в низком логическом уровне и переводится в высокий уровень по окончании приема последнего бита блока данных, как и в режиме передачи одного слова.

1.6.20 Формат синхронного обмена Microwire фирмы National Semiconductor

Временные диаграммы последовательного синхронного обмена в режиме Microwire показывает Рисунок 68 (одиночный обмен) и Рисунок 69 (непрерывный обмен).

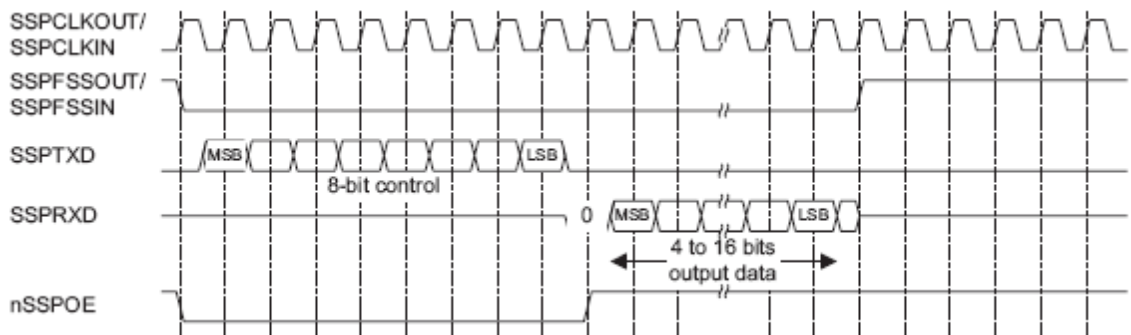


Рисунок 68 – Формат синхронного обмена протокола Microwire (одиночный обмен)

Протокол передачи данных Microwire во многом схож с протоколом SPI, за исключением того, что обмен в нем осуществляется в полудуплексном режиме, с использованием служебных последовательностей. Каждая информационный обмен начинается с передачи ведущим устройством специальной 8-битной управляющей последовательности. В течение всего времени ее передачи приемник не обрабатывает каких-либо входных данных. После того, как сигнал передан и декодирован ведомым устройством, оно выдерживает паузу в один тактовый интервал после передачи последнего бита управляющей последовательности, после чего передает в адрес ведущего устройства запрошенные данные. Длительность блока данных от ведомого устройства может составлять от 4 до 16 бит, таким образом, общая длительность информационного кадра составляет от 13 до 25 бит.

В данном режиме во время ожидания приемопередатчика:

- сигнал SSPCLK имеет низкий логический уровень;
- сигнал SSPFSS имеет высокий логический уровень;
- сигнал SSPTXD переводится в высокоимпедансное состояние.

Переход в режим информационного обмена происходит после записи управляющего байта в буфер FIFO передатчика. По заднему фронту сигнала SSPFSS данные из буфера переносятся в регистр сдвига блока передатчика, откуда, начиная со старшего значащего разряда, последовательно выдаются в линию SSPTXD. Линия SSPFSS остается в низком логическом уровне в течение всей передачи кадра. Линия SSPRXD при этом находится в высокоимпедансном состоянии.

Внешнее ведомое устройство осуществляет прием бит данных по переднему фронту сигнала SSPCLK. По окончании приема последнего бита управляющей последовательности она декодируется в течение одного тактового интервала, после чего ведомое устройство передает запрошенные данные в адрес модуля SSP. Биты данных выдаются в линию SSPRXD по заднему фронту сигнала SSPCLK. Ведущее устройство в свою очередь регистрирует их по переднему фронту этого тактового сигнала. В случае одиночного информационного обмена по окончании приема последнего бита слова данных сигнал SSPFSS переводится в высокий уровень на время, соответствующее одному тактовому интервалу, что служит командой для переноса принятого слова данных из регистра сдвига в буфер FIFO приемника.

Примечание – Внешнее устройство может перевести линию приемника в третье состояние по заднему фронту сигнала SSPCLK после приема последнего бита слова данных, либо после перевода линии SSPFSS в высокий логический уровень.

Непрерывный обмен данными начинается и заканчивается так же, как и одиночный обмен. Однако линия SSPFSS удерживается в низком логическом уровне в течение всего сеанса передачи данных. Управляющий байт следующего информационного кадра передается сразу же после приема младшего значащего разряда текущего кадра. Данные из сдвигового регистра передаются в буфер приемника после регистрации младшего разряда очередного слова по заднему фронту сигнала SSPCLK.

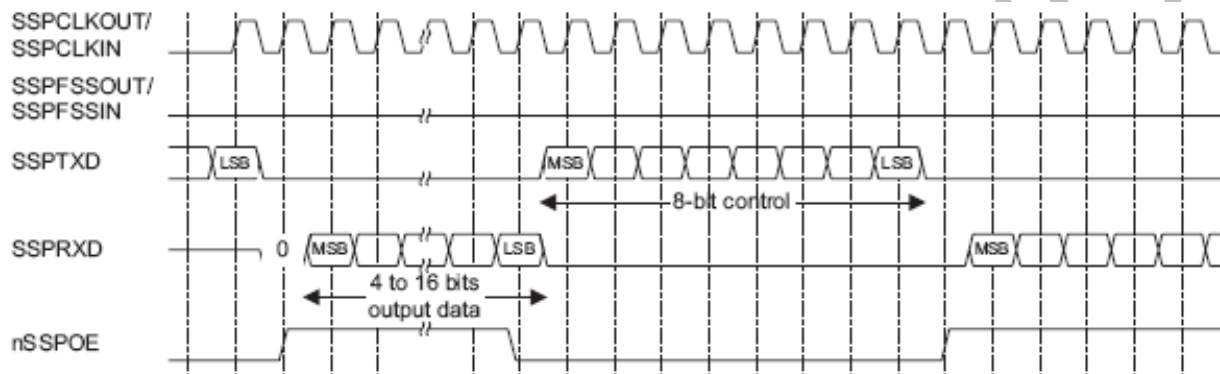


Рисунок 69 – Формат синхронного обмена протокола Microwire (непрерывный обмен)

Требования к временным параметрам сигнала SSPFSS относительно тактового сигнала SSPCLK в режиме Microwire

Модуль SSP, работающий в режиме Microwire как ведомое устройство, регистрирует данные по переднему фронту сигнала SSPCLK после установки сигнала SSPFSS в низкий логический уровень. Ведущие устройства, формирующие сигнал SSPCLK, должны гарантировать достаточное время установки и удержания сигнала SSPFSS по отношению к переднему фронту сигнала SSPCLK.

Данные требования иллюстрирует Рисунок 70. По отношению к переднему фронту сигнала SSPCLK, по которому осуществляется регистрация данных в приемнике ведомого модуля SSP, время установки сигнала SSPFSS должно быть как минимум в два раза больше периода SSPCLK, на котором работает модуль. По отношению к предыдущему переднему фронту сигнала SSPCLK должно обеспечиваться время удержания не менее одного периода этого тактового сигнала.

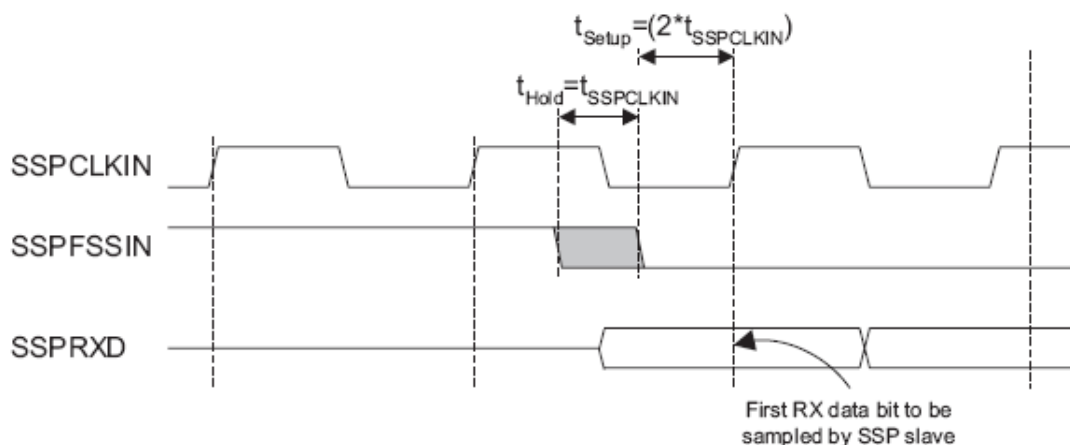


Рисунок 70 – Формат Microwire, требования к времени установки и удержания сигнала SSPFSSIN

1.6.21 Примеры конфигурации модуля в ведущем и ведомом режимах

Рисунок 71, Рисунок 72, Рисунок 73 показывают варианты подключения модуля PrimeCell SSP (PL022) к периферийным устройствам, работающим в ведущем или ведомом режиме.

Примечание – Модуль SSP не поддерживает динамическое изменение режима ведущий/ведомый. Каждый приемопередатчик должен быть изначально сконфигурирован в одном из этих режимов.

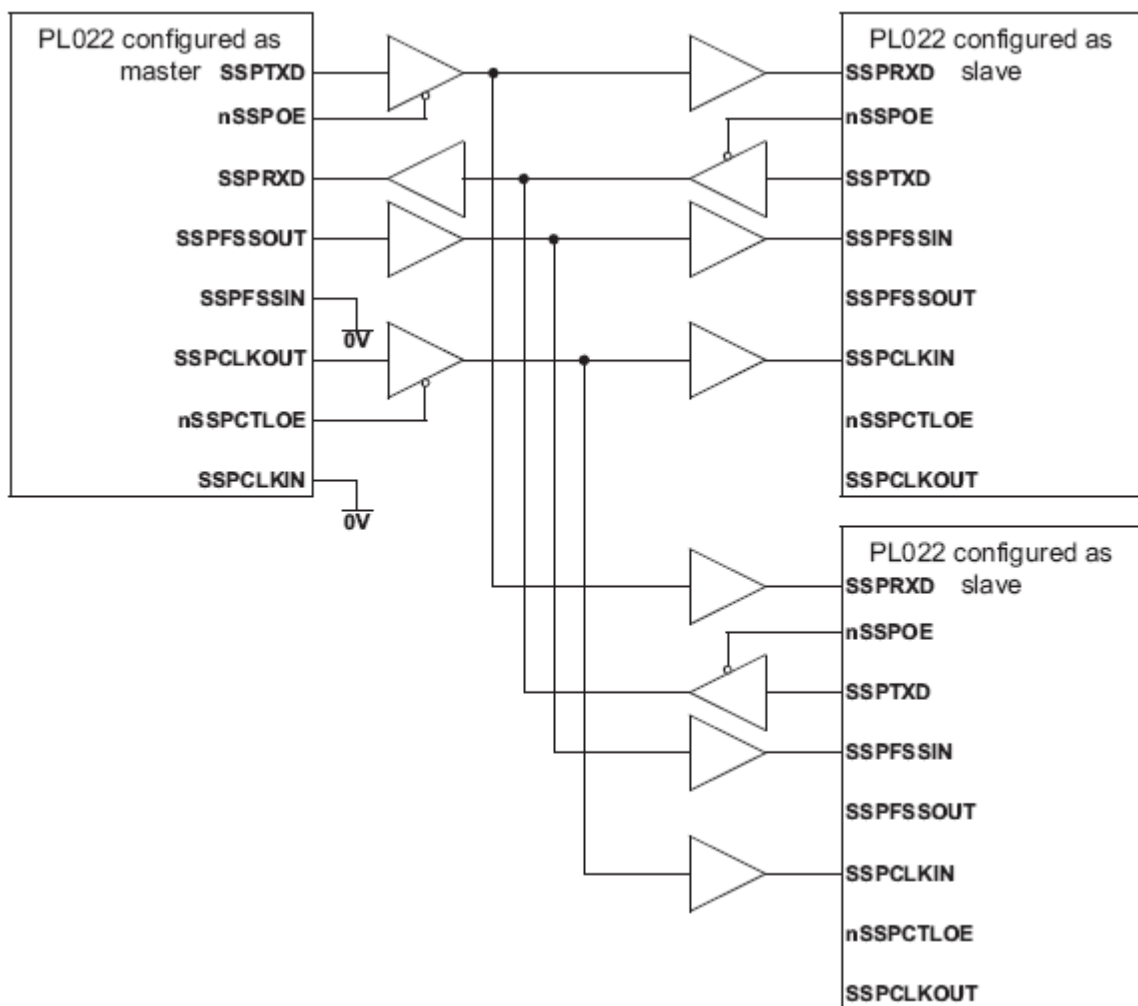


Рисунок 71 – Ведущее устройство SSP подключено к двум ведомым

Рисунок 71 показывает совместную работу трех модулей SSP, один из которых сконфигурирован в качестве ведущего, а два – в качестве ведомых устройств. Ведущее устройство способно передавать данные циркулярно в адрес двух ведомых по линии SSPTXD.

Для ответной передачи данных один из ведомых модулей разрешает прохождение сигнала от своей линии SSPTXD на вход SSPRXD ведущего.

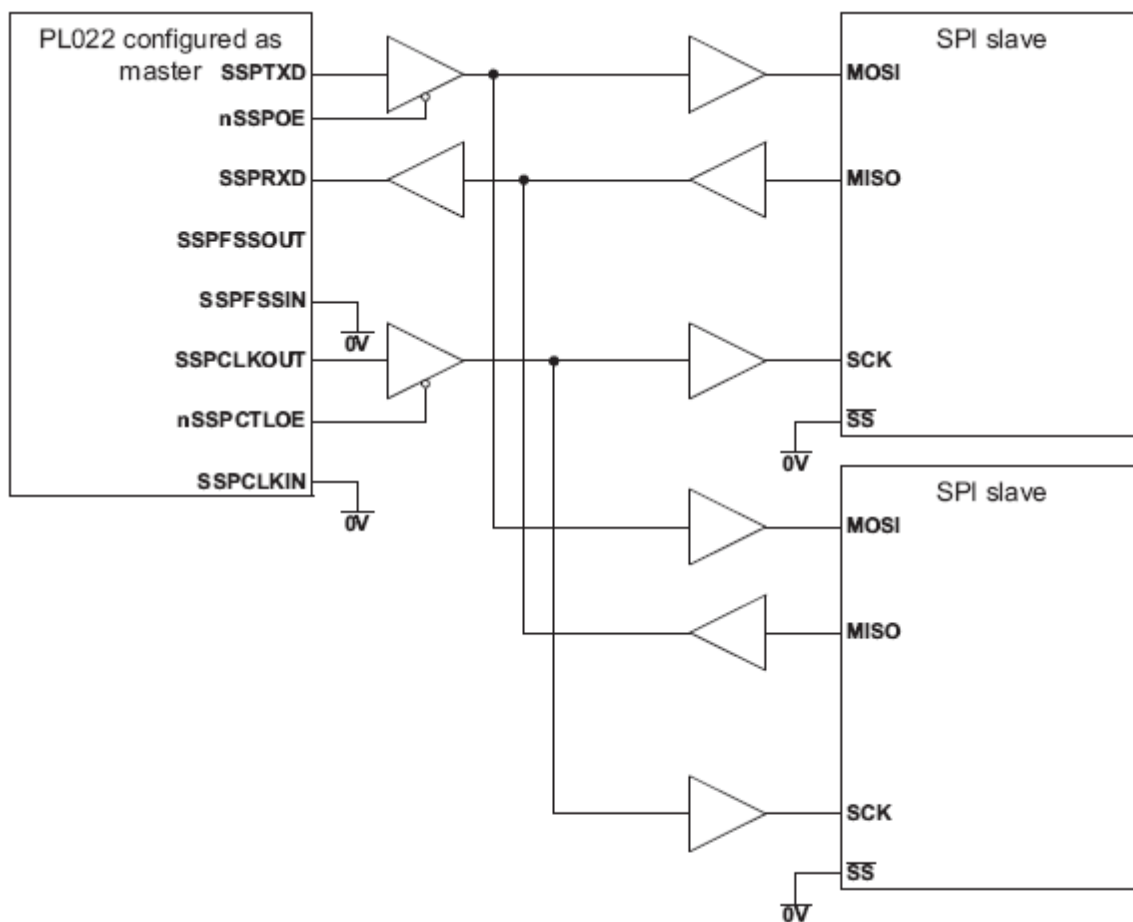


Рисунок 72 – Ведущее устройство SSP подключено к двум ведомым, поддерживающим SPI

Рисунок 72 показывает подключение модуля SSP, сконфигурированного как ведущее устройство, к двум ведомым устройствам, поддерживающим протокол SPI фирмы Motorola. Внешние устройства сконфигурированы как ведомые путем установки в низкий логический уровень сигнала выбора ведомого устройства Slave Select (SS). Как и в предыдущем примере, ведущее устройство способно передавать данные в адрес ведомых циркулярно по линии SSPTXD. Ответная передача данных на входную линию SSPRXD ведущего устройства одновременно осуществляется только одним из ведомых по соответствующей линии MISO.

1.7 Интерфейс прямого доступа к памяти

Модуль SSP предоставляет интерфейс подключения к контроллеру прямого доступа к памяти. Работа в данном режиме контролируется регистром управления DMA DMACR.

Интерфейс DMA включает в себя следующие сигналы:

Для приема:

- SSPRXDMASREQ – запрос передачи отдельного слова, инициируется приемопередатчиком. Сигнал переводится в активное состояние в случае, если буфер FIFO приемника содержит, по меньшей мере, одно слово;
- SSPRXDMABREQ – запрос блочного обмена данными, инициируется модулем приемопередатчика. Сигнал переходит в активное состояние в случае, если буфер FIFO приемника содержит четыре или более слов;
- SSPRXDMACLR – сброс запроса на DMA, инициируется контроллером DMA с целью сброса принятого запроса. В случае если был запрошен блочный обмен данными, сигнал сброса формируется в ходе передачи последнего слова данных в блоке.

Для передачи:

- SSPTXDMASREQ – запрос передачи отдельного слова, инициируется модулем приемопередатчика. Сигнал переводится в активное состояние в случае, если буфер FIFO передатчика содержит, по меньшей мере, одну свободную ячейку;
- SSPTXDMABREQ – запрос блочного обмена данными, инициируется модулем приемопередатчика. Сигнал переводится в активное состояние в случае, если буфер FIFO передатчика содержит четыре или менее слов;
- SSPTXDMACLR – сброс запроса на DMA, инициируется контроллером DMA с целью сброса принятого запроса. В случае если был запрошен блочный обмен данными, сигнал сброса формируется в ходе передачи последнего слова данных в блоке.

Сигналы блочного и одноэлементного обмена данными не являются взаимоисключающими, они могут быть инициированы одновременно. Например, в случае, если заполнение данными буфера приемника превышает пороговое значение четыре, формируются как сигнал запроса одноэлементного обмена, так и сигнал запроса блочного обмена данными. В случае если количество данных в буфере приема меньше порогового значения, формируется только запрос одноэлементного обмена. Это бывает полезно в ситуациях, при которых объем данных меньше размера блока. Пусть, например, нужно принять 19 символов. Тогда контроллер DMA осуществит четыре передачи блоков по четыре символа, а оставшиеся три символа передаст в ходе трех одноэлементных обменов.

Примечание – Для оставшихся трех символов контроллер SSP не инициирует процедуру блочного обмена.

Каждый инициированный приемопередатчиком сигнал запроса DMA остается активным до момента его сброса соответствующим сигналом DMACLR.

После снятия сигнала сброса модуль приемопередатчика вновь получает возможность сформировать запрос на DMA в случае выполнения описанных выше условий. Все запросы

DMA снимаются после запрета работы приемопередатчика, а также в случае снятия сигнала разрешения DMA.

Таблица 351 показывает значения порогов заполнения буферов приемника и передатчика, необходимых для срабатывания запросов блочного обмена DMABREQ.

Таблица 351 – Параметры срабатывания запросов блочного обмена данными в режиме DMA

Пороговый уровень	Длина блока обмена данными	
	Буфер передатчика (количество незаполненных ячеек)	Буфер приемника (количество заполненных ячеек)
$\frac{1}{2}$	4	4

Рисунок 74 показывает временные диаграммы одноэлементного и блочного запросов DMA, в том числе действие сигнала DMACLR. Все сигналы должны быть синхронизированы с PCLK.

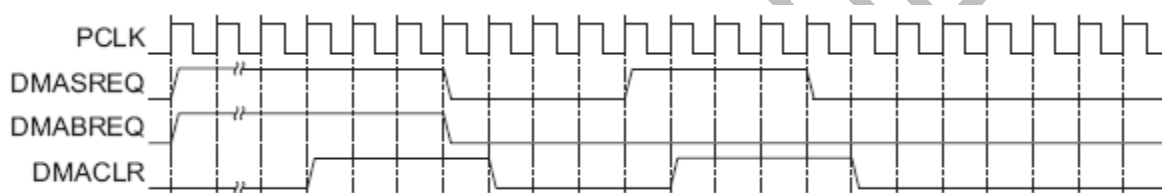


Рисунок 74 – Временные диаграммы обмена в режиме DMA

1.8 Программное управление модулем

1.8.1 Общая информация

Следующие адреса являются резервными и не должны использоваться в нормальном режиме функционирования:

- адреса со смещениями в диапазоне +0x028 ... +0x07C и +0xFD0 ... +0xFDC зарезервированы для перспективных расширений возможностей модуля;
- адреса со смещениями в диапазоне +0x080 ... +0x088 зарезервированы для тестирования.

1.8.2 Описание регистров контроллера SSP

Данные о регистрах модуля SSP приведены в таблице ниже (Таблица 352).

Таблица 352 – Обобщенные данные о регистрах модуля SSP

Базовый Адрес	Наименование				Описание
0x4000_0000	SSP1				Регистры контроллера интерфейса SSP1
0x400C_8000	SS2				Регистры контроллера интерфейса SSP2
0x400D_0000	SSP3				Регистры контроллера интерфейса SSP3

Смещение	Наименование	Тип	Значение после сброса	Размер, бит	Описание
0x000	CR0	RW	0x0000	16	Регистр управления 0
0x004	CR1	RW	0x0	4	Регистр управления 1
0x008	DR	RW	0x----	16	Буфера FIFO приемника (чтение) Буфер FIFO передатчика (запись)
0x00C	SR	RO	0x03	3	Регистр состояния
0x010	CPSR	RW	0x00	8	Регистр делителя тактовой частоты
0x014	IMSC	RW	0x0	4	Регистр маски прерывания
0x018	RIS	RO	0x8	4	Регистр состояния прерываний без учета маскирования
0x01C	MIS	RO	0x0	4	Регистр состояния прерываний с учетом маскирования
0x020	ICR	WO	0x0	4	Регистр сброса прерывания
0x024	DMACR	RW	0x0	2	Регистр управления прямым доступом к памяти

Примечание – В поле «тип» указан вид доступа к регистру: RW – чтение и запись, RO – только чтение, WO – только запись.

Регистр управления 0 CR0

Регистр CR0 содержит пять битовых полей, предназначенных для управления блоками модуля SSP. Назначение разрядов регистра представлено в таблице ниже (Таблица 353).

Таблица 353 – Формат регистра CR0

Бит	Наименование	Назначение
15...8	SCR	Скорость последовательного обмена. Значение поля SCR используется при формировании тактового сигнала обмена данными. Информационная скорость удовлетворяет соотношению: $SSPCLK / (CPSDVR * (1 + SCR))$, где CPSDVR – четное число в диапазоне от 2 до 254 (см. регистр SSPCPSR), а SCR – число от 0 до 255
7	SPH	Фаза сигнала SSPCLK (используется только в режиме обмена SPI фирмы Motorola). См. раздел «Формат SPI фирмы Motorola»
6	SPO	Полярность сигнала SSPCLK (используется только в режиме обмена SPI фирмы Motorola). См. раздел «Формат SPI фирмы Motorola»
5...4	FRF	Формат информационного кадра. 00 – протокол SPI фирмы Motorola; 01 – протокол SSI фирмы Texas Instruments; 10 – протокол Microwire фирмы National Semiconductor; 11 – резерв
3...0	DSS	Размер слова данных. 0000 – резерв. 0001 – резерв. 0010 – резерв. 0011 – 4 бита. 0100 – 5 бит. 0101 – 6 бит.

		0110 – 7 бит. 0111 – 8 бит. 1000 – 9 бит. 1001 – 10 бит. 1010 – 11 бит. 1011 – 12 бит. 1100 – 13 бит. 1101 – 14 бит. 1110 – 15 бит. 1111 – 16 бит
--	--	--

ПРЕДВАРИТЕЛЬНО

Регистр управления 1 CR1

Регистр CR1 содержит четыре битовых поля, предназначенных для управления блоками модуля SSP. Назначение разрядов регистра представлено в таблице ниже (Таблица 354).

Таблица 354 – Регистр CR1

Биты	Наименование	Назначение
15...4		Зарезервировано. При чтении результат не определен. При записи следует устанавливать в 0.
3	SOD	Запрет выходных линий в режиме ведомого устройства. Бит используется только в режиме ведомого устройства (MS=1). Это позволяет организовать двусторонний обмен данными в системах, содержащих одно ведущее и несколько ведомых устройств. Бит SOD следует установить в случае, если данный ведомый модуль SSP не должен в настоящее время осуществлять передачу данных в линию SSPTXD. При этом линии обмена данных ведомых устройств можно соединить параллельно. 0 – управление линией SSPTXD в ведомом режиме разрешена. 1 – управление линией SSPTXD в ведомом режиме запрещена.
2	MS	Выбор ведущего или ведомого режима работы: 0 – ведущий модуль (устанавливается по умолчанию); 1 – ведомый модуль.
1	SSE	Разрешение работы приемопередатчика: 0 – работа запрещена; 1 – работа разрешена.
0	LBM	Тестирование по шлейфу: 0 – нормальный режим работы приемопередатчика; 1 – выход регистра сдвига передатчика соединен с входом регистра сдвига приемника.

Регистр данных DR

Регистр DR имеет разрядность 16 бит и предназначен для чтения принятых и записи передаваемых данных.

Операция чтения обеспечивает доступ к последней несчитанной ячейке буфера FIFO приемника. Запись данных в этот буфер FIFO осуществляет блок приемника.

Операция записи позволяет занести очередное слово в буфер FIFO передатчика. Извлечение данных из этого буфера осуществляет блок передатчика. При этом извлеченные данные помещаются в регистр сдвига передатчика, откуда последовательно выдаются на линию SSPTXD с заданной скоростью информационного обмена.

В случае если выбран размер информационного слова менее 16 бит, перед записью в регистр DR необходимо обеспечить выравнивание данных по правой границе. Блок передатчика игнорирует неиспользуемые биты. Принятые информационные слова автоматически выравниваются по правой границе в блоке приемника.

В режиме обмена данными Microwire фирмы National Semiconductor модуль SSP по умолчанию работает с восьмиразрядными информационными словами (старший значащий байт игнорируется). Размер принимаемых данных задается программно. Буфера FIFO приемника и передатчика автоматически не очищаются даже в случае, если бит SSE сброшен в 0. Это позволяет заполнить буфер передатчика необходимой информацией заблаговременно, перед разрешением работы модуля.

Назначение разрядов регистра DR описано в таблице ниже (Таблица 355).

Таблица 355 – Формат регистра DR

Бит	Наименование	Назначение
15...0	DATA	Принимаемые данные (чтение) Передаваемые данные (запись) В случае если выбран размер информационного слова менее 16 бит, перед записью в регистр DR необходимо обеспечить выравнивание данных по правой границе. Блок передатчика игнорирует неиспользуемые биты. Принятые информационные слова автоматически выравниваются по правой границе в блоке приемника.

Регистр состояния SR

Регистр состояния доступен только для чтения и содержит информацию о состоянии буферов FIFO приемника и передатчика и занятости модуля SSP.

Таблица 356 показывает назначение бит регистра SR.

Таблица 356 – Регистр SR

Биты	Наименование	Назначение
15...5	-	Зарезервировано. При чтении результат не определен. При записи следует заполнить нулями.
4	BSY	Флаг занятости модуля: 0 – модуль SSP неактивен; 1 – модуль SSP в настоящее время передает и/или принимает данные, либо буфер FIFO передатчика не пуст.
3	RFF	Буфер FIFO приемника заполнен: 0 – не заполнен; 1 – заполнен.
2	RNE	Буфер FIFO приемника не пуст: 0 – пуст; 1 – не пуст.
1	TNF	Буфер FIFO передатчика не заполнен: 0 – заполнен; 1 – не заполнен.
0	TFE	Буфер FIFO передатчика пуст: 0 – не пуст; 1 – пуст.

Регистр делителя тактовой частоты CPSR

Регистр CPSR используется для установки параметров делителя тактовой частоты. Записываемое значение должно быть целым числом в диапазоне от 2 до 254. Младший

значащий разряд регистра принудительно устанавливается в ноль. Если записать в регистр CPSR нечетное число, его последующее чтение даст результатом это число, но с установленным в ноль младшим битом.

Назначение бит регистра CPSR представлено в таблице ниже (Таблица 357).

Таблица 357 – Регистр CPSR

Биты	Наименование	Назначение
15...8	-	Зарезервировано. При чтении результат не определен. При записи следует заполнить нулями.
7...0	CPSDVSР	Коэффициент деления тактовой частоты. Записываемое значение должно быть целым числом в диапазоне от 2 до 254. Младший значащий разряд регистра принудительно устанавливается в ноль.

Регистр установки и сброса маски прерывания IMSC

При чтении выдается текущее значение маски. При записи производится установка или сброс маски на соответствующее прерывание. При этом запись 1 в разряд разрешает соответствующее прерывание, запись 0 – запрещает.

После сброса все биты регистра маски устанавливаются в нулевое состояние.

Назначение битов регистра IMSC показано в таблице ниже (Таблица 358).

Таблица 358 – Регистр IMSC

Биты	Наименование	Назначение
15...4	-	Зарезервировано. При чтении выдаются нули. При записи следует заполнить нулями.
3	TXIM	Маска прерывания по заполнению наполовину и менее буфера FIFO передатчика. 1 – не маскирована; 0 – маскирована.
2	RXIM	Маска прерывания по заполнению наполовину и более буфера FIFO приемника. 1 – не маскирована; 0 – маскирована.
1	RTIM	Маска прерывания по таймауту приемника (буфер FIFO приемника не пуст и не было попыток его чтения в течение времени таймаута). 1 – не маскирована, 0 – маскирована.
0	RORIM	Маска прерывания по переполнению буфера приемника. 1 – не маскирована, 0 – маскирована.

Регистр состояния прерываний RIS

Этот регистр доступен только для чтения и содержит текущее состояние прерываний без учета маскирования. Данные, записываемые в регистр, игнорируются.

Назначение бит в регистре RIS представлено в таблице ниже (Таблица 359).

Таблица 359 – Регистр RIS

Биты	Наименование	Назначение
15...4	-	Зарезервировано. При чтении выдаются нули
3	TXRIS	Состояние до маскирования прерывания SSPTXINTR. 1 – буфер FIFO передатчика заполнен наполовину или менее; 0 – буфер FIFO передатчика заполнен более чем наполовину
2	RXRIS	Состояние до маскирования прерывания SSPRXINTR. 1 – буфер FIFO приемника заполнен наполовину или более; 0 – буфер FIFO приемника заполнен менее чем наполовину
1	RTRIS	Состояние до маскирования прерывания SSPRTINTR. 1 – истекло время таймаута приемника; 0 – время таймаута приемника не истекло
0	RORRIS	Состояние до маскирования прерывания SSPRORINTR. 1 – возникло событие переполнения буфера приемника; 0 – событие переполнения буфера приемника не возникало

Регистр маскированного состояния прерываний MIS

Этот регистр доступен только для чтения и содержит текущее состояние прерываний с учетом маскирования. Данные, записываемые в регистр, игнорируются.

Назначение бит в регистре MIS представлено в таблице ниже (Таблица 360).

Таблица 360 – Регистр MIS

Биты	Наименование	Назначение
15...4	-	Зарезервировано. При чтении выдаются нули.
3	TXMIS	Состояние маскированного прерывания SSPTXINTR. 1 – буфер FIFO передатчика заполнен наполовину или менее; 0 – буфер FIFO передатчика заполнен более чем наполовину.
2	RXMIS	Состояние маскированного прерывания SSPRXINTR. 1 – буфер FIFO приемника заполнен наполовину или более; 0 – буфер FIFO приемника заполнен менее чем наполовину.
1	RTMIS	Состояние маскированного прерывания SSPRTINTR. 1 – истекло время таймаута приемника; 0 – время таймаута приемника не истекло.
0	RORMIS	Состояние маскированного прерывания SSPRORINTR. 1 – возникло событие переполнения буфера приемника; 0 – событие переполнения буфера приемника не возникало.

Регистр сброса прерываний ICR

Этот регистр доступен только для записи и предназначен для сброса признака прерывания по заданному событию путем записи 1 в соответствующий бит. Запись в любой из разрядов регистра 0 игнорируется.

Назначение бит в регистре ICR представлено в таблице ниже (Таблица 361).

Таблица 361 – Регистр ICR

Биты	Наименование	Назначение
15...2	-	Зарезервировано. При записи следует заполнить нулями.
1	RTIC	Сброс прерывания SSPRTINTR.
0	RORIC	Сброс прерывания SSPRORINTR.

Регистр управления прямым доступом к памяти DMACR

Регистр доступен по чтению и записи. После сброса все биты регистра обнуляются.

Назначение бит регистра DMACR представлено таблице ниже (Таблица 362).

Таблица 362 – Регистр DMACR

Биты	Наименование	Назначение
15...2	-	Зарезервировано. При чтении выдаются нули. При записи следует заполнить нулями.
1	TXDMAE	Использование DMA при передаче. 1 – разрешено формирование запросов DMA для обслуживания буфера FIFO передатчика; 0 – запрещено формирование запросов DMA для обслуживания буфера FIFO передатчика.
0	RXDMAE	Использование DMA при приеме. 1 – разрешено формирование запросов DMA для обслуживания буфера FIFO приемника; 0 – запрещено формирование запросов DMA для обслуживания буфера FIFO приемника.

1.9 Прерывания

В модуле предусмотрено пять маскируемых линий запроса на прерывание, в том числе, четыре независимые линии запроса с активным высоким логическим уровнем, а также один общий сигнал, представляющий собой комбинацию независимых по схеме ИЛИ.

Сигналы запроса на прерывание:

- SSPRXINTR – запрос на обслуживание буфера FIFO приемника.
- SSPTXINTR – запрос на обслуживание буфера FIFO передатчика.
- SSPRORINTR – переполнение буфера FIFO приемника.
- SSPRTINTR – таймаут приемника.
- SSPINTR – логическое ИЛИ сигналов SSPRXINTR, SSPTXINTR, SSPRTINTR и SSPRORINTR.

Каждый из независимых сигналов запроса на прерывание может быть маскирован путем установки соответствующего бита в регистре маски SSPx_IMSC. Установка бита в 1 разрешает соответствующее прерывание, в 0 – запрещает.

Доступность как индивидуальных, так и общей линии запроса позволяет организовать обслуживание прерываний в системе, как путем применения глобальной процедуры обработки, так и с помощью драйвера устройства, построенного по модульному принципу.

Прерывания от приемника и передатчика SSPRXINTR и SSPTXINTR выведены отдельно от прерываний по изменению состояния устройства. Это позволяет использовать данные сигналы запроса для обеспечения чтения и записи данных, согласованной с достижением заданного порога заполнения буферов FIFO приемника и передатчика.

Признаки возникновения каждого из условий прерывания можно считать из регистра прерываний SSPx_RIS или из маскированного регистра прерываний SSPx_MIS.

1.9.1 SSPRXINTR

Прерывание по заполнению буфера FIFO приемника формируется в случае, если буфер приемника содержит половину или более несчитанных слов данных.

1.9.2 SSPTXINTR

Прерывание по заполнению буфера FIFO передатчика формируется в случае, если буфер передатчика содержит четыре или менее корректных слов данных.

Состояние прерывания не зависит от значения сигнала разрешения работы модуля SSP. Это позволяет организовать взаимодействие программного обеспечения с передатчиком одним из двух способов. Во-первых, можно записать данные в буфер заблаговременно, перед активизацией передатчика и разрешения прерываний. Во-вторых, можно предварительно разрешить работу модуля и формирование прерываний и заполнять буфер передатчика в ходе работы процедуры обслуживания прерываний.

1.9.3 SSPRORINTR

Прерывание по переполнению буфера FIFO приемника формируется в случае, если буфер уже заполнен и блоком приемника осуществлена попытка записать в него еще одно слово. При этом принятое слово данных регистрируется в регистре сдвига приемника, но в буфер приемника не заносится.

1.9.4 SSPRTINTR

Прерывание по таймауту приемника возникает в случае, если буфер FIFO приемника не пуст, и на вход приемника не поступало новых данных в течение периода времени, необходимого для передачи 32 бит. Данный механизм гарантирует, что пользователь будет знать о наличии в буфере приемника необработанных данных.

Прерывание по таймауту снимается либо после считывания данных из буфера приемника до его опустошения, либо после приема новых слов данных по входной линии SSPRXD. Кроме того, оно может быть снято путем записи 1 в бит RTIC регистра сброса прерывания SSPx_ICR.

1.9.5 SSPINTR

Все описанные сигналы запроса на прерывание скомбинированы в общую линию путем объединения по схеме ИЛИ сигналов SSPRXINTR, SSPTXINTR, SSPRTINTR и SSPRORINTR с учетом маскирования. Общий выход может быть подключен к системному контроллеру прерывания, что позволит ввести дополнительное маскирование запросов на уровне периферийных устройств.

ПРЕДВАРИТЕЛЬНО

Контроллер UART

Модуль универсального асинхронного приемопередатчика (UART – Universal Asynchronous Receiver-Transmitter) представляет собой периферийное устройство микроконтроллера.

В состав контроллера включен кодек (ENDEC – Encoder/Decoder) последовательного интерфейса инфракрасной (ИК) передачи данных в соответствии с протоколом SIR (SIR – Serial Infra Red) ассоциации Infrared Data Association (IrDA).

1.1 Основные характеристики модуля UART

Модуль UART может быть запрограммирован для использования, как в качестве универсального асинхронного приемопередатчика, так и для инфракрасного обмена данными (SIR).

Модуль содержит независимые буферы приема (16 слов по 13 бит) и передачи (16 слов по 9 бит) типа FIFO (First In First Out – первый вошел, первый вышел), что позволяет снизить интенсивность прерываний центрального процессора.

Программное отключение FIFO позволяет ограничить размер буфера одним словом.

Есть возможность программно настраивать скорость обмена данными, путем деления тактовой частоты опорного генератора в диапазоне ($1 \times 16 - 65535 \times 16$). Допускается использование нецелых коэффициентов деления частоты, что позволяет использовать любой опорный генератор с частотой более 3,6864 МГц.

Модулем поддерживаются стандартные элементы асинхронного протокола связи – стартового и стопового бит, а также бита контроля четности, которые добавляются перед передачей и удаляются после приема.

Независимо могут быть маскированы прерывания от буфера FIFO передатчика, буфера FIFO приемника, по таймауту приемника, по изменению линий состояния модема, а также в случае обнаружения ошибки.

Модуль UART позволяет использовать DMA-контроллер для организации обмена данными и снижения нагрузки на ядро микроконтроллера.

Также модуль обеспечивает обнаружение ложных стартовых бит.

Формирование и обнаружения сигнала разрыва линии.

Возможность организации аппаратного управления потоком данных.

Полностью программируемый UART имеет следующие характеристики:

- данные длиной 5, 6, 7, 8 или 9 бит;
- формирование и контроль четности (проверочный бит выставляется по четности, нечетности, имеет фиксированное значение, либо не передается);
- формирование 1 или 2 стоповых бит;
- скорость передачи данных – до $UART_CLK/16$ Бод.

Кодек ИК-обмена данными IrDA SIR обеспечивает:

- программный выбор обмена данными по линиям асинхронного приемопередатчика либо кодекса ИК связи IrDA SIR;
- поддержку функционирования с информационной скоростью до 115200 бит/с в режиме полудуплекса;
- поддержку длительности бит для нормального режима (3/16) и для режима пониженного энергопотребления (1,41 – 2,23 мкс);
- программируемое деление опорной частоты $UART_CLK$ для получения заданной длительности бит в режиме пониженного энергопотребления.

ПРЕДВАРИТЕЛЬНО

1.2 Программируемые параметры

Следующие ключевые параметры могут быть заданы программно:

- скорость передачи данных – целая и дробная часть числа;
- количество бит данных;
- количество стоповых бит;
- режим контроля четности;
- разрешение или запрет использования буферов FIFO (глубина очереди данных – 16 элементов или один элемент, соответственно);
- порог срабатывания прерывания по заполнению буферов FIFO (1/8, 1/4, 1/2, 3/4 и 7/8);
- частота внутреннего тактового генератора (номинальное значение – 1,8432 МГц) может быть задана в диапазоне 1,42 – 2,12 МГц для обеспечения возможности формирования бит данных с укороченной длительностью в режиме пониженного энергопотребления (для ИК-обмена);
- режим аппаратного управления потоком данных.

1.3 Отличия от контроллера UART 16C650

Контроллер отличается от промышленного стандарта асинхронного приемопередатчика 16C650 следующими характеристиками:

- пороги срабатывания прерывания по заполнению буфера FIFO приемника – 1/8, 1/4, 1/2, 3/4 и 7/8;
- пороги срабатывания прерывания по заполнению буфера FIFO передатчика – 1/8, 1/4, 1/2, 3/4 и 7/8;
- отличается распределение адресов внутренних регистров и назначение бит в регистрах;
- недоступны изменения сигналов состояния модема.

Следующие возможности контроллера 16C650 не поддерживаются:

- полуторная длительность стопового бита (поддерживается только 1 или 2 стоповых бита);
- независимое задание тактовой частоты приемника и передатчика.

1.4 Функциональные возможности

Устройство выполняет следующие функции:

- преобразование данных, полученных от периферийного устройства, из последовательной в параллельную форму;
- преобразование данных, передаваемых на периферийное устройство, из параллельной в последовательную форму.

Процессор читает и записывает данные, а также управляющую информацию и информацию о состоянии модуля. Прием и передача данных буферизуются с помощью внутренней памяти FIFO, позволяющей сохранить до 16 слов по 13 бит для режима приема и до 16 слов по 9 бит для режима передачи.

Модуль приемопередатчика:

- содержит программируемый генератор, формирующий тактовый сигнал одновременно для передачи и для приема данных на основе внутреннего тактового сигнала UART_CLK;
- обеспечивает возможности, сходные с возможностями индустриального стандарта – контроллера UART 16C650;
- позволяет осуществлять обмен информацией с максимальной скоростью:
 - в режиме IrDA – до 460800 бит/с;
 - в режиме IrDA с пониженным энергопотреблением – до 115200 бит/с.

Режим работы приемопередатчика и скорость обмена данными контролируются регистром управления линией LCR_H регистрами делителя скорости передачи данных – целой части (IBRD) и дробной части (FBRD).

Устройство может формировать следующие сигналы:

- независимые маскируемые прерывания от приемника (в том числе по таймауту), передатчика, а также по изменению состояния модема и в случае обнаружения ошибки;
- общее прерывание, возникающее в случае, если возникло одно из независимых немаскированных прерываний;
- сигналы запроса на прямой доступ к памяти (DMA) для совместной работы с контроллером DMA.

В случае возникновения ошибки в структуре сигнала, четности данных, а также разрыва линии соответствующий бит ошибки устанавливается и сохраняется в буфере FIFO. В случае переполнения буфера немедленно устанавливается соответствующий бит в регистре переполнения, а доступ к записи в буфер FIFO блокируется.

Существует возможность программно ограничить размер буфера FIFO одним байтом, что позволяет реализовать общепринятый интерфейс асинхронной последовательной связи с двойной буферизацией.

Поддерживаются входные линии состояния модема: «готовность к приему» (Clear To Send, CTS), «обнаружен информационный сигнал» (Data Carrier Detected, DCD), «источник данных готов» (Data Set Ready, DSR) и «индикатор вызова» (Ring Indicator, RI), а также выходные линии: «запрос на передачу» (Request to Send, RTS) и «приемник данных готов» (Data Terminal Ready, DTR).

Доступна возможность аппаратного управления потоком данных по линиям nUARTCTS и nUARTRTS.

Блок последовательного интерфейса инфракрасной передачи данных в соответствии с протоколом IrDA SIR реализует протокол обмена данными ENDEC. В случае его активизации обмен информацией осуществляется не с помощью сигналов UARTTXD и UARTRXD, а посредством сигналов nSIROUT и SIRIN.

В этом случае устройство переводит линию UARTTXD в пассивное состояние (высокий уровень), и перестает реагировать на изменение состояния модема, а также сигнала на линии UARTRXD. Протокол SIR ENDEC обеспечивает возможность обмена данными исключительно в режиме полудуплекса, то есть он не может передавать во время приема данных и принимать во время передачи данных.

В соответствии со спецификацией физического уровня протокола IrDA SIR, задержка между передачей и приемом должна составлять не менее 10 мс.

1.5 Описание функционирования блока UART

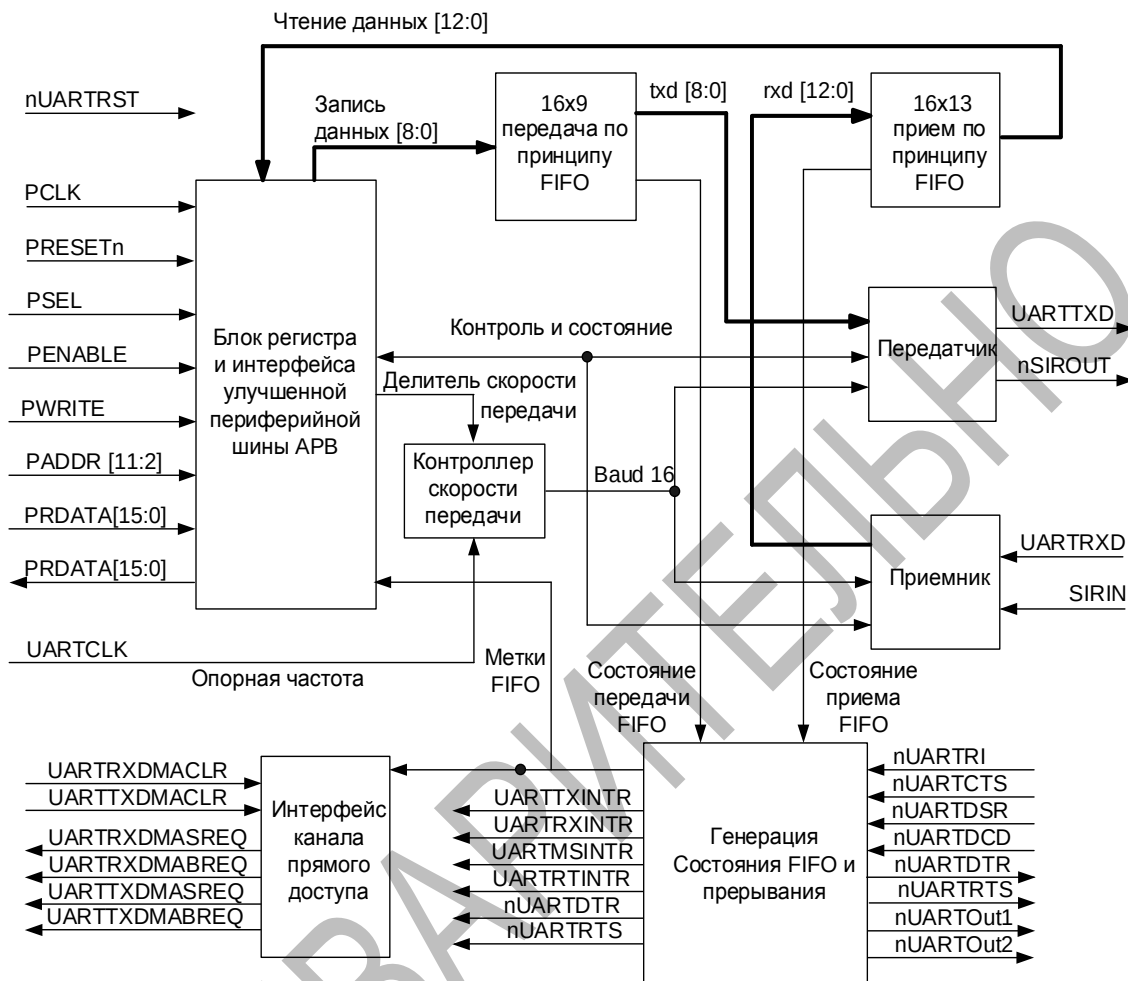


Рисунок 75 – Блок-схема универсального асинхронного приёмопередатчика (UART)

1.5.1 Генератор тактового сигнала приемопередатчика

Генератор содержит счетчики без цепи сброса, формирующие внутренние тактовые сигналы $Baud16$ и $IrLPBaud16$.

Сигнал $Baud16$ используется для синхронизации схем управления приемником и передатчиком последовательного обмена данными. Он представляет собой последовательность импульсов с шириной, равной одному периоду сигнала $UART_CLK$ и частотой, в 16 раз выше скорости передачи данных.

Сигнал $IrLPBaud16$ предназначен для синхронизации схемы формирования импульсов с длительностью, требуемой для ИК обмена данными в режиме с пониженным энергопотреблением.

1.5.2 Буфер FIFO передатчика

Буфер передатчика имеет ширину 9 бит, глубину 16 слов, схему организации доступа типа FIFO («первый вошел, первый вышел»). Данные от центрального процессора, записанные через шину APB, сохраняются в буфере до тех пор, пока не будут считаны

логической схемой передачи данных. Существует возможность запретить буфер FIFO передатчика, в этом случае он будет функционировать как однобайтовый буферный регистр.

1.5.3 Буфер FIFO приемника

Буфер приемника имеет ширину 13 бит, глубину 16 слов, схему организации доступа типа FIFO («первый вошел, первый вышел»). Принятые от периферийного устройства данные и соответствующие коды ошибок сохраняются логикой приема данных в нем до тех пор, пока не будут считаны центральным процессором через шину APB. Буфер FIFO приемника может быть запрещен, в этом случае он будет действовать как однобайтовый буферный регистр.

1.5.4 Блок передатчика

Логические схемы передатчика осуществляют преобразование данных, считанных из буфера передатчика, из параллельной в последовательную форму. Управляющая логика выдает последовательный поток бит в порядке: стартовый бит, биты данных, начиная с младшего значащего разряда, бит проверки на четность, и, наконец, стоповые биты, в соответствии с конфигурацией, записанной в регистре управления.

1.5.5 Блок приемника

Логические схемы приемника преобразуют данные, полученные от периферийного устройства, из последовательной в параллельную форму после обнаружения корректного стартового импульса. Кроме того, производятся проверки на: переполнение буфера, ошибки контроля четности, ошибки в структуре сигнала и на разрыв линии. Признаки обнаружения этих ошибок также сохраняются в выходном буфере.

1.5.6 Блок формирования прерываний

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

Комбинированный сигнал прерывания может быть подан на внешний контроллер прерываний системы, при этом появится дополнительная возможность маскирования устройства в целом, что облегчает построение модульных драйверов устройств.

Другой подход состоит в подаче на системный контроллер прерываний независимых сигналов запроса на прерывание от приемопередатчика. В этом случае процедура обработки сможет одновременно считать информацию обо всех источниках прерываний. Данный подход привлекателен в случае, если скорость доступа к регистрам периферийных устройств значительно превышает тактовую частоту центрального процессора в системе реального времени.

Для более подробной информации см. раздел Прерывания.

1.5.7 Интерфейс прямого доступа к памяти

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

1.5.8 Блок и регистры синхронизации

Контроллер поддерживает как асинхронный, так и синхронный режимы работы тактовых генераторов CPU_CLK и UART_CLK. Регистры синхронизации и логика

квитирования постоянно находятся в активном состоянии. Это практически не отражается на характеристиках устройства и занимаемой площади. Синхронизация сигналов управления осуществляется в обоих направлениях потока данных, то есть как из области действия CPU_CLK в область действия UART_CLK, так и наоборот.

1.6 Описание функционирования ИК кодека IrDA SIR

Структурная схема кодека представлена на рисунке 113.

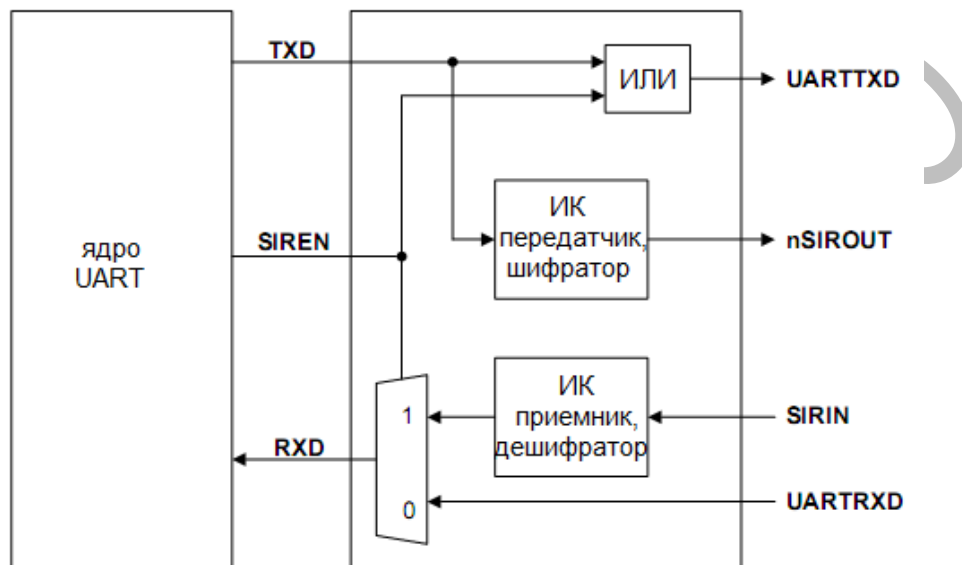


Рисунок 76 – Структурная схема кодека IrDA

1.6.1 Кодер ИК передатчика

Кодер преобразует поток данных с выхода асинхронного передатчика, сформированный по закону модуляции без возврата к нулю (NRZ). Спецификация физического уровня протокола IrDA SIR подразумевает использование модуляции с возвратом к нулю и инверсией (RZI), в соответствии с которой передача логического нуля соответствует излучению одного светового ИК импульса. Сформированный выходной поток импульсов подается на усилитель и далее на ИК светодиод.

Длительность импульса в режиме IrDA составляет, согласно спецификации, 3 периода внутреннего тактового генератора с частотой Baud16, то есть 3/16 периода времени, выделенного на передачу одного бита.

В режиме IrDA с пониженным энергопотреблением ширина импульса задана как 3/16 периода, выделенного на передачу бита, при скорости передачи данных 115200 бит/с. Данное требование реализуется за счет формирования трех периодов тактового сигнала IrLPBaud16 с номинальной частотой 1,8432 МГц, в свою очередь, формируемого путем деления частоты UART_CLK. Значение частоты IrLPBaud16 задается путем записи соответствующего коэффициента деления частоты в регистр ILPR.

Выход кодера имеет активное низкое состояние. При передаче логической единицы выход кодера остается в низком состоянии, при передаче логического нуля – формируется импульс, при этом выход кратковременно переводится в высокое состояние.

Как в нормальном режиме, так и в режиме пониженного энергопотребления, использование нецелых значений коэффициента деления скорости передачи данных увеличивает джиттер («дребезжание») фронтов импульсов данных. Наличие джиттера в

случае использования дробных коэффициентов деления связано с тем, что интервалы между тактовыми импульсами Baud16 будут нерегулярными – период сигнала Baud16 в разное время будет содержать различное количество периодов сигнала UART_CLK. Можно показать, что в наихудшем случае величина джиттера в потоке ИК импульсов может достигать трех периодов UART_CLK. В соответствии со спецификацией стандарта IrDA SIR, джиттер не должен превышать величины 13 %. В случае, если частота сигнала UART_CLK составляет более 3,6834 МГц, а скорость передачи данных меньше или равна 115200 бит/с, величина джиттера не превышает 9 %. Таким образом, требования стандарта выполняются.

ПРЕДВАРИТЕЛЬНО

1.6.2 Декодер ИК приемника

Декодер преобразует поток данных, сформированных по закону возврата к нулю, полученного от приемника ИК сигнала, и выдает поток данных без возврата к нулю на вход приемника UART. В неактивном состоянии вход декодера находится в высоком состоянии. Выходной сигнал кодера имеет полярность, противоположную полярности входа декодера.

Обнаружение стартового бита осуществляется при низком уровне сигнала на входе декодера.

Примечание – Для того чтобы исключить ложные срабатывания UART от импульсных помех, на входе SIRIN игнорируются импульсы с длительностью менее, чем:

- 3/16 длительности Baud16 в режиме IrDA;
- 3/16 длительности IrLPBaud16 в режиме IrDA с пониженным энергопотреблением.

1.7 Описание работы UART

1.7.1 Сброс модуля

Приемопередатчик и кодек могут быть сброшены общим сигналом сброса процессора. Значения регистров после сброса описаны в подразделе «Программное управление модулем».

1.7.2 Тактовые сигналы

Частота тактового сигнала F_{UART_CLK} должна обеспечивать поддержку требуемого диапазона скоростей передачи данных:

$$F_{UART_CLK}(\min) \geq 16 * \text{baud_rate_max};$$
$$F_{UART_CLK}(\max) \leq 16 * 65535 * \text{baud_rate_min}.$$

Например, для поддержки скорости передачи данных в диапазоне от 110 до 460800 Бод частота F_{UART_CLK} должна находиться в интервале от 7,3728 МГц до 115,34 МГц.

Частота F_{UART_CLK} , кроме того, должна выбираться с учетом возможности установки скорости передачи данных в рамках заданных требований точности.

Также существует ограничение на соотношение между тактовыми частотами F_{CPU_CLK} и F_{UART_CLK} . Частота F_{UART_CLK} должна быть не более чем в 5/3 раз выше частоты F_{CPU_CLK} .

$$F_{UART_CLK} \leq (5/3) * F_{CPU_CLK}.$$

Например, при работе в режиме UART с максимальной скоростью передачи данных 921600 бод, при частоте $F_{UART_CLK} = 14,7456$ МГц, частота F_{CPU_CLK} должна быть не менее 8,85276 МГц. Это гарантирует, что контроллер UART будет иметь достаточно времени для записи принятых данных в буфер FIFO.

1.7.3 Работа универсального асинхронного приемопередатчика

Управляющая информация хранится в регистре управления линией LCR. Этот регистр имеет внутреннюю ширину 30 бит, однако внешний доступ по шине APB к нему осуществляется через следующие регистры:

- LCR_H – определяет:
 - параметры передачи данных;
 - длину слова;
 - режим буферизации;
 - количество передаваемых стоповых бит;
 - режим контроля четности;
 - формирование сигнала разрыва линии;
- IBRD – определяет целую часть коэффициента деления для скорости передачи данных;
- FBRD – определяет дробную часть коэффициента деления для скорости передачи данных.

Примечание – Изменение значений трех регистров можно осуществить корректно двумя способами:

- запись IBRD, запись FBRD, запись LCR_H;
- запись FBRD, запись IBRD, запись LCR_H.

Чтобы изменить значение лишь одного из регистров (IBRD или FBRD), необходимо выполнить следующие шаги:

- запись IBRD (или FBRD), запись LCR_H.

1.7.4 Коэффициент деления частоты

Коэффициент деления для формирования скорости передачи данных состоит из 22 бит, при этом 16 бит выделено для представления его целой части, а 6 бит – дробной части. Возможность задания нецелых коэффициентов деления позволяет осуществлять обмен данными со стандартными информационными скоростями не превышающими 230400, при этом используя в качестве UART_CLK тактовый сигнал с произвольной частотой более 3,6864 МГц.

Целая часть коэффициента деления записывается в 16-битный регистр IBRD. Шестиразрядная дробная часть записывается в регистр FBRD. Значение коэффициента деления связано с содержимым указанных регистров следующим образом:

Коэффициент деления = $F_{UART_CLK} / (16 * \text{скорость передачи данных}) = IBRD + FBRD$,
где

IBRD – целая часть, а FBRD – дробная часть коэффициента деления.

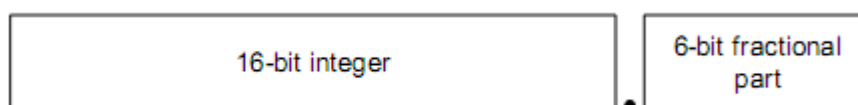


Рисунок 77 – Коэффициент деления

Шестибитное значение, записываемое в регистр FBRD, вычисляется путем выделения дробной части требуемого коэффициента деления, умножения ее на 64 (то есть на 2^n , где n – ширина регистра FBRD) и округления до ближайшего целого числа:

$$M = \text{integer}(\text{FBRD} * 2^n + 0.5),$$

где

integer – операция отсечения дробной части числа, $n = 6$.

В модуле формируется внутренний сигнал Baud16, представляющий собой последовательность импульсов с длительностью, равной периоду сигнала UART_CLK и средней частотой, в 16 раз большей требуемой скорости обмена данными.

1.7.5 Передача и прием данных

Принятые или передаваемые данные заносятся в 16-элементные буферы FIFO, при этом каждый элемент приемного буфера FIFO кроме байта данных хранит также четыре бита информации о состоянии модема.

Данные для передачи заносятся в буфер FIFO передатчика. Если работа приемопередатчика разрешена, начинается передача информационного кадра с параметрами, указанными в регистре управления линией LCR_H. Передача данных продолжается до опустошения буфера FIFO передатчика. После записи элемента в буфер FIFO передатчика сигнал BUSY переходит в высокое состояние. Это состояние сохраняется в течение всего времени передачи данных. В низкое состояние сигнал BUSY переходит только после того, как буфер FIFO передатчика станет пуст, а последний бит данных (включая стоповые биты) будет передан. Сигнал BUSY может находиться в высоком состоянии даже в случае, если приемопередатчик будет переведен из разрешенного состояния в запрещенное.

Для каждого бита данных (в приемной линии) производится три измерения уровня, решение принимается по мажоритарному принципу.

В случае если приемник находился в неактивном состоянии (на линии входного сигнала UARTRXD постоянно присутствует единица) и произошел переход входного сигнала из высокого в низкий логический уровень (обнаружен стартовый бит), включается счетчик, тактируемый сигналом Baud16, после чего отсчеты сигнала на входе приемника регистрируются каждые восемь тактов (в режиме асинхронного приемопередатчика) или каждые четыре такта (в режиме ИК обмена данными) сигнала Baud16. Более частая выборка данных в режиме ИК обмена связана с необходимостью корректной обработки импульсов данных согласно протоколу SIR IrDA.

Стартовый бит считается достоверным в случае, если сигнал на линии UARTRXD сохраняет низкий логический уровень в течение восьми отсчетов сигнала Baud16 с момента включения счетчика. В противном случае переход в ноль рассматривается как ложный старт и игнорируется.

В случае если обнаружен достоверный стартовый бит, производится регистрация последовательности данных на входе приемника. Очередной бит данных фиксируются каждые 16 отсчетов тактового сигнала Baud16 (что соответствует длительности одного бита символа). Производится регистрация всех бит данных (согласно запрограммированным параметрам) и бита четности (если включен режим контроля четности).

Наконец, производится проверка присутствия корректного стопового бита (высокий логический уровень сигнала UARTRXD). В случае если последнее условие не выполняется, устанавливается признак ошибки формирования кадра. После того, как слово данных

принято полностью, оно заносится в буфер FIFO приемника, наряду с четырьмя битами признаков ошибки, связанных с принятым словом.

1.7.6 Биты ошибки

Три бита признаков ошибки, ассоциированные с принятым символом данных, заносятся на позиции [11:9] слова данных в буфере FIFO приемника. Также предусмотрен признак ошибки переполнения буфера FIFO, расположенный на позиции 12 слова данных.

Таблица 363 показывает назначение всех битов слова данных в FIFO буфере приемника.

Таблица 363 – Назначение бит слова данных в FIFO-буфере приемника

Бит буфера FIFO	Назначение
12	Признак переполнения буфера
11	Ошибка – разрыв линии
10	Ошибка проверки на четность
09	Ошибка формирования кадра
08...00	Принятые данные

1.7.7 Бит переполнения буфера

Бит переполнения непосредственно не связан с конкретным символом в буфере приемника. Признак переполнения фиксируется в случае, если буфер FIFO заполнен к моменту, когда очередной символ данных полностью принят (находится в регистре сдвига). При этом данные из регистра сдвига не попадают в буфер приемника и теряются с началом приема очередного символа. Как только в буфере приемника появляется свободное место, очередной принятый символ данных заносится в буфер FIFO вместе с текущим значением признака переполнения. После успешной записи данных в буфер признак переполнения сбрасывается.

1.7.8 Запрет буфера FIFO

Предусмотрена возможность отключения FIFO буферов приемника и передатчика. В этом случае приемная и передающая сторона контроллера UART располагают лишь однобайтными буферными регистрами. Бит переполнения буфера устанавливается при этом тогда, когда очередной символ данных уже принят, однако предыдущий еще не был считан.

В настоящей реализации модуля буферы FIFO физически не отключаются, необходимая функциональность достигается за счет логических манипуляций с флагами. При этом в случае, если буфер FIFO отключен, а сдвиговый регистр передатчика пуст (не используется), запись байта данных происходит непосредственно в регистр сдвига, минуя буферный регистр.

Проверка по шлейфу

Проверка по шлейфу (замыкание выхода передатчика на вход приемника) выполняется путем установки в 1 бита LBE в регистре управления контроллером CR.

ПРЕДВАРИТЕЛЬНО

1.7.9 Работа кодека ИК обмена данными IrDA SIR

Кодек обеспечивает сопряжение асинхронного потока данных, сформированного приемопередатчиком, с полудуплексным последовательным интерфейсом IrDA SIR. Какая-либо аналоговая обработка сигнала при этом не выполняется. Назначение кодека – сформировать цифровой поток данных на вход приемника асинхронного сигнала и обработать цифровой поток данных с выхода передатчика.

Предусмотрено два режима работы:

В режиме IrDA уровень логического нуля передается на линию nSIROUT в виде импульса с высоким логическим уровнем и длительностью 3/16 от выбранного периода следования бит данных. Логическая единица при этом передается в виде постоянного низкого уровня сигнала. Сформированный выходной сигнал далее подается на передатчик ИК сигнала, обеспечивая излучение светового импульса всякий раз при передаче нулевого бита. На приемной стороне световые импульсы воздействуют на базу фототранзистора ИК приемника, который в результате формирует низкий логический уровень. Это, в свою очередь, обуславливает низкий уровень на входе SIRIN.

В режиме IrDA с пониженным энергопотреблением длительность передаваемых импульсов ИК излучения устанавливается в три раза больше длительности импульсов внутреннего опорного сигнала IrLPBaud16 (равной 1.63 мкс при номинальной частоте 1.8432 МГц). Данный режим активизируется путем установки бита SIR_LP в регистре управления CR.

Как в нормальном режиме, так и в режиме пониженного энергопотребления:

- кодирование осуществляется на основе бит данных, сформированных асинхронным передатчиком модуля;
- в ходе приема данных декодированные биты далее обрабатываются блоком асинхронного приема.

В соответствии со спецификацией физического уровня протокола IrDA SIR, обмен данными должен осуществляться в режиме полудуплекса, при этом задержка между передачей и приемом данных должна составлять не менее 10 мс. Эта задержка должна формироваться программно. Необходимость ее введения обусловлена тем, что воздействие передающего ИК светодиода на находящийся рядом ИК приемник может привести к искажению принимаемого сигнала или даже ввести приемный тракт в состояние насыщения. Задержка между окончанием передачи и началом приема данных именуется латентность, или время установки (готовности) приемника.

Сигнал IrLPBaud16 формируется путем деления частоты сигнала UART_CLK в соответствии с коэффициентом деления, записанным в регистре ILPR.

Коэффициент деления вычисляется по формуле:

$$\text{Коэффициент деления} = F_{\text{UART_CLK}} / \text{IrLPBaud16},$$

где номинальное значение IrLPBaud16 составляет 1,8432 МГц. Коэффициент деления должен быть выбран так, чтобы выполнялось соотношение:

$$1,42 \text{ МГц} < \text{IrLPBaud16} < 2,12 \text{ МГц}.$$

Проверка по шлейфу

Проверка по шлейфу выполняется после установки в 1 бита LBE регистра управления контроллером CR с одновременной установкой в 1 бита SIRTEST регистра управления тестированием TCR.

В этом режиме данные, передаваемые на выход nSIROUT, должны подаваться на вход SIRIN.

Примечание – Это единственный случай использования тестового регистра в нормальном режиме функционирования модуля.

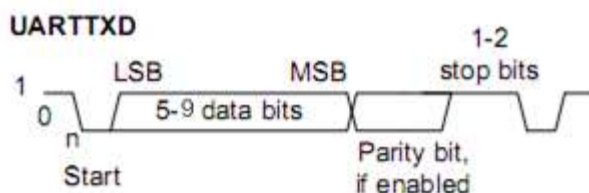


Рисунок 78 – Кадр передачи данных

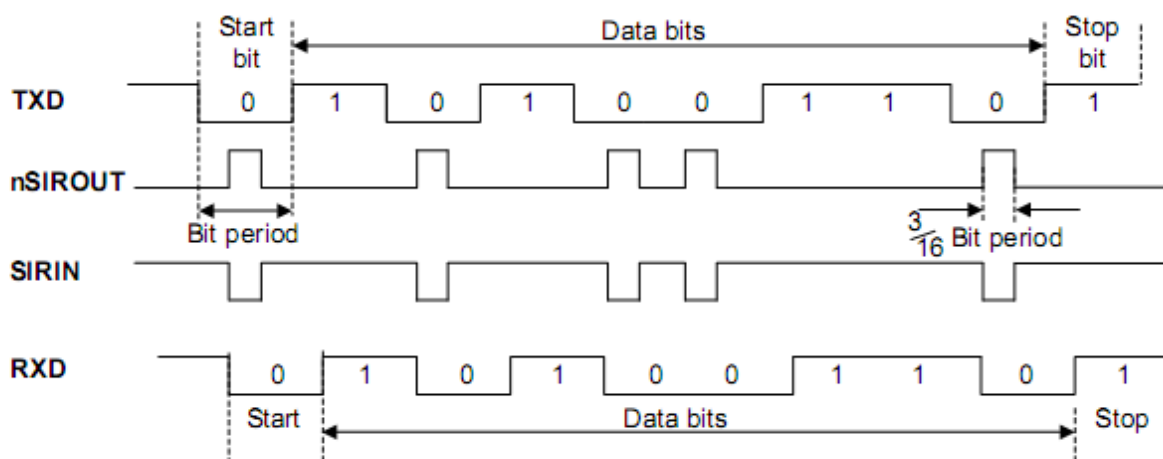


Рисунок 79 – Модуляция данных IrDA

1.8 Линии управления модемом

Модуль универсального асинхронного приемопередатчика может использоваться как в режиме оконечного оборудования (DTE), так и в режиме оборудования передачи данных (DCE).

Таблица 364 – Назначение управления модемом в режимах DTE и DCE

Сигнал	Назначение	
	Режим оконечного оборудования	Режим оборудования передачи данных
nUARTCTS	Готов к передаче данных	Запрос передачи данных
nUARTDSR	Источник данных готов	Приемник данных готов
nUARTDCD	Обнаружен информационный сигнал	-
nUARTRI	Индикатор вызова	-
nUARTRTS	Запрос передачи данных	Готов к передаче данных
nUARTDTR	Приемник данных готов	Источник данных готов
nUARTOut1	-	Обнаружен информационный сигнал
nUARTOut2	-	Индикатор вызова

1.8.1 Аппаратное управление потоком данных

Программно активизируемый режим аппаратного управления потоком данных позволяет контролировать (приостанавливать и возобновлять) информационный обмен с помощью сигналов nUARTRTS и nUARTCTS. Иллюстрация взаимодействия двух устройств последовательной связи с аппаратным управлением потоком данных представлена на рисунке ниже.

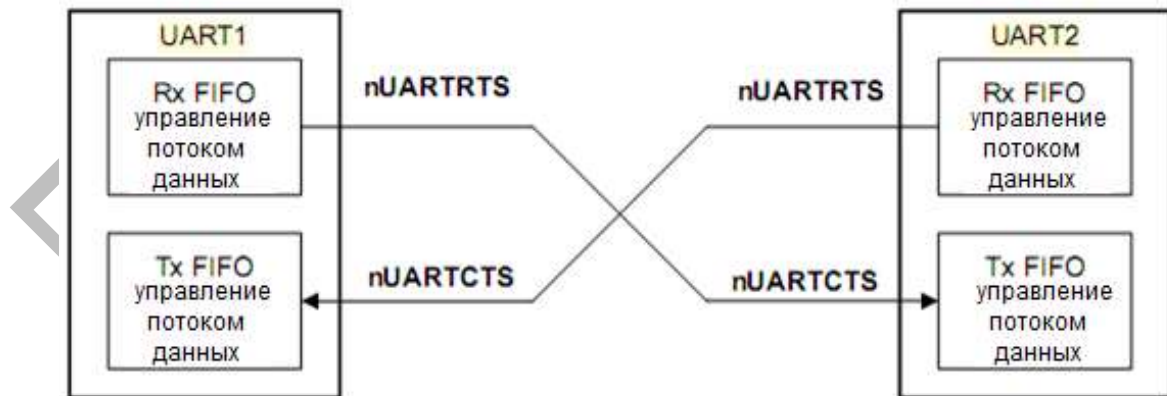


Рисунок 80 – Взаимодействие двух устройств последовательной связи с аппаратным управлением потоком данных

Если разрешено управление потоком данных по сигналу RTS, линия nUARTRTS переводится в активное состояние только после того, как в FIFO буфере приема появляется заданное количество свободных элементов.

Если разрешено управление потоком данных по сигналу CTS, передача данных осуществляется только после перевода линии nUARTCTS в активное состояние.

Режим аппаратного управления потоком данных задается путем установки значений бит RTSEn и CTSEn в регистре управления CR. В таблице 363 показаны необходимые установки для различных режимов управления потоком данных.

Таблица 365 – Режимы управления потоком данных

CTSEn	RTSEn	Описание
1	1	Разрешено управление потоком данных по CTS и RTS
1	0	Управления потоком данных осуществляется по линии CTS
0	1	Управления потоком данных осуществляется по линии RTS
0	0	Управления потоком данных запрещено

Примечание – В случае если выбран режим управления потоком данных по RTS, программное обеспечение не может использовать бит RTS регистра CR для проверки состояния линии RTS.

1.8.2 Управление потоком данных по линии RTS

Логика управления потоком данных по RTS использует данные о превышении пороговых уровней заполнения буфера FIFO приемника. В случае выбора режимов с управлением по RTS, сигнал на линии nUARTRTS переводится в активное состояние только после того, как в FIFO буфере приема появляется заданное количество свободных элементов. После достижения порогового уровня заполнения буфера приемника сигнал nUARTRTS снимается (переводится в пассивное состояние), указывая, таким образом, на отсутствие свободного места для сохранения принятых данных. При этом дальнейшая передача данных должна быть прекращена по завершении передачи текущего символа.

Обратно в активное состояние сигнал nUARTRTS переводится после считывания данных из приемного буфера FIFO в количестве, достаточном для того, чтобы заполнение буфера оказалось ниже порогового уровня.

В случае если управление потоком данных по RTS запрещено, однако работа приемопередатчика UART разрешена, прием будет осуществляться до полного заполнения буфера FIFO, либо до завершения передачи данных.

1.8.3 Управление потоком данных по линии CTS

В случае выбора одного из режимов с управлением потоком данных по CTS передатчик осуществляет проверку состояния линии nUARTCTS перед началом передачи очередного байта данных. Передача осуществляется только в случае, если данная линия активна, и продолжается до тех пор, пока активное состояние линии сохраняется и буфер передатчика не пуст.

При переходе линии nUARTCTS в неактивное состояние модуль завершает выдачу текущего передаваемого символа, после чего передача данных прекращается.

Если управление потоком данных по CTS запрещено, и при этом работа приемопередатчика UART разрешена – данные будут выдаваться до опустошения буфера FIFO передатчика.

ПРЕДВАРИТЕЛЬНО

1.9 Интерфейс прямого доступа к памяти

Модуль универсального асинхронного приемопередатчика оснащен интерфейсом подключения к контроллеру прямого доступа к памяти. Работа в данном режиме контролируется регистром управления DMA DMACR.

Интерфейс DMA включает в себя следующие сигналы:

Для приема:

UARTRXDMASREQ – запрос передачи отдельного символа, инициируется контроллером UART. Размер символа в режиме приема данных – до 13 бит. Сигнал переводится в активное состояние в случае, если буфер FIFO приемника содержит по меньшей мере один символ.

UARTRXDMABREQ – запрос блочного обмена данными, инициируется модулем приемопередатчика. Сигнал переходит в активное состояние в случае, если заполнение буфера FIFO приемника превысило заданный порог. Порог программируется индивидуально для каждого буфера FIFO путем записи значения в регистр IFLS.

UARTRXDMACLR – сброс запроса на DMA, инициируется модулем приемопередатчика с целью сброса принятого запроса. В случае если был запрошен блочный обмен данными, сигнал сброса формируется в ходе передачи последнего символа данных в блоке.

Для передачи:

UARTTXDMASREQ – запрос передачи отдельного символа, инициируется модулем приемопередатчика. Размер символа в режиме передачи данных – до 9 бит. Сигнал переводится в активное состояние в случае, если буфер FIFO передатчика содержит, по меньшей мере, одну свободную ячейку.

UARTTXDMABREQ – запрос блочного обмена данными, инициируется модулем приемопередатчика. Сигнал переводится в активное состояние в случае, если заполнение буфера FIFO передатчика ниже заданного порога. Порог программируется индивидуально для каждого буфера FIFO путем записи значения в регистр IFLS.

UARTTXDMACLR – сброс запроса на DMA, инициируется контроллером DMA с целью сброса принятого запроса. В случае если был запрошен блочный обмен данными, сигнал сброса формируется в ходе передачи последнего символа данных в блоке.

Сигналы блочного и одноэлементного обмена данными не являются взаимно исключающими, они могут быть инициированы одновременно. Например, в случае, если заполнение данными буфера приемника превышает пороговое значение, формируется как сигнал запроса одноэлементного обмена, так и сигнал запроса блочного обмена данными. В случае если количество данных в буфере приема меньше порогового значения формируется только запрос одноэлементного обмена. Это бывает полезно в ситуациях, при которых объем данных меньше размера блока. Пусть, например, нужно принять 19 символов, а порог заполнения буфера FIFO установлен равным четырем. Тогда контроллер DMA осуществит четыре передачи блоков по четыре символа, а оставшиеся три символа передаст в ходе трех одноэлементных обменов.

Примечание – Для оставшихся трех символов контроллер UART не может инициировать процедуру блочного обмена.

Каждый инициированный приемопередатчиком сигнал запроса DMA остается активным до момента его сброса соответствующим сигналом DMACLR.

После снятия сигнала сброса модуль приемопередатчика вновь получает возможность сформировать запрос на DMA в случае выполнения описанных выше условий. Все запросы DMA снимаются после запрета работы приемопередатчика, а также в случае установки в ноль бита управления DMA TXDMAE или RXDMAE в регистре управления DMA DMACR.

В случае запрета буферов FIFO устройство способно передавать и принимать только одиночные символы, как следствие, контроллер может инициировать DMA только в одноэлементном режиме. При этом модуль в состоянии формировать только сигналы управления DMA UARTRXDMASREQ и UARTTXDMASREQ. Для информации о запрете буферов FIFO см. описание регистра управления линией LCR_H.

Когда буферы FIFO включены, обмен данными может производиться в ходе как одноэлементных, так и блочных передач данных, в зависимости от установленной величины порога заполнения буферов и их фактического заполнения. Таблица 366 показывает значения параметров срабатывания запросов блочного обмена UARTRXDMABREQ и UARTTXDMABREQ в зависимости от порога заполнения буфера.

Таблица 366 – Параметры срабатывания запросов блочного обмена данными в режиме DMA

Пороговый уровень	Длина блока обмена данными	
	Буфер передатчика (количество незаполненных ячеек)	Буфер приемника (количество заполненных ячеек)
1/8	14	2
1/4	12	4
1/2	8	8
3/4	4	12
7/8	2	14

В регистре управления DMA DMACR предусмотрен бит DMAONERR, который позволяет запретить DMA от приемника в случае активного состояния линии прерывания по обнаружению ошибки UARTEINTR. При этом соответствующие линии запроса DMA: UARTRXDMASREQ и UARTRXDMABREQ переводятся в неактивное состояние (маскируются) до сброса UARTEINTR. На линии запроса DMA, обслуживающие передатчик, состояние UARTEINTR не влияет.

Рисунок 81 показывает временные диаграммы одноэлементного и блочного запросов DMA, в том числе действие сигнала DMACLR. Все сигналы должны быть синхронизированы с CPU_CLK. В интересах ясности изложения предполагается, что синхронизация сигналов запроса DMA в контроллере DMA не производится.

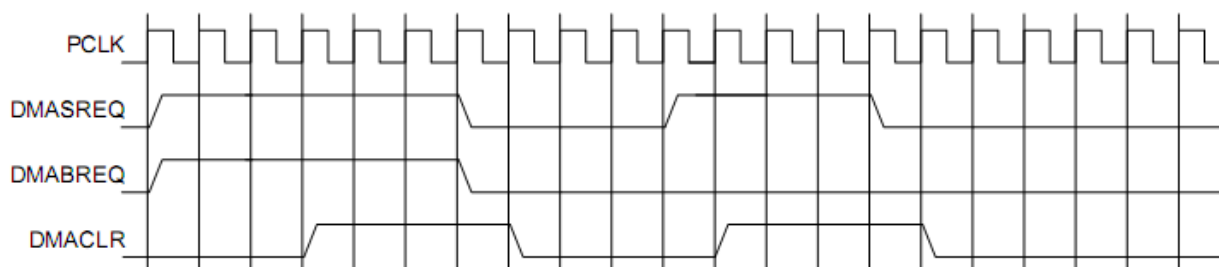


Рисунок 81 – Временные диаграммы одноэлементного и блочного запросов DMA

ПРЕДВАРИТЕЛЬНО

1.10 Прерывания

В модуле предусмотрено 11 маскируемых источников прерывания. В результате формируется один общий сигнал, представляющий собой комбинацию независимых сигналов, объединенных по схеме ИЛИ.

Сигналы запроса на прерывание:

- UARTRXINTR – прерывание от приемника;
- UARTRXINTR – прерывание от передатчика;
- UARTRNEINTR – FIFO приемника не пусто;
- UARTRFEINTR – FIFO передатчика пусто;
- UARTRNBSYINTR – сдвиговый регистр передатчика пуст;
- UARTRTINTR – прерывание по таймауту приемника;
- UARTRMSINTR – прерывание по состоянию модема:
 - UARTRIINTR, изменение состояния линии nUARTRI;
 - UARTRCTSINTR, изменение состояния линии nUARTRCTS;
 - UARTRDCDINTR, изменение состояния линии nUARTRDCD;
 - UARTRDSRINTR, изменение состояния линии nUARTRDSR.
- UARTEINTR – ошибка:
 - UARTRTOEINTR, переполнение буфера;
 - UARTRBEINTR, прерывание приема – разрыв линии;
 - UARTRPEINTR, ошибка контроля четности;
 - UARTRFEINTR, ошибка в структуре кадра.
- UARTRINTR – логическое ИЛИ сигналов UARTRXINTR, UARTRTXINTR, UARTRTINTR, UARTRMSINTR, UARTRTEINTR, UARTRRNEINTR, UARTRTFEINTR и UARTRTNBSYINTR.

Каждый из независимых сигналов запроса на прерывание может быть маскирован путем установки соответствующего бита в регистре маски UARTRMSC. Установка бита в 1 разрешает соответствующее прерывание, в 0 – запрещает.

Доступность, как индивидуальных линий, так и общей линии запроса позволяет организовать обслуживание прерываний в системе, как путем применения глобальной процедуры обработки, так и с помощью драйвера устройства, построенного по модульному принципу.

Прерывания от приемника и передатчика UARTRXINTR и UARTRTXINTR выведены отдельно от прерываний по изменению состояния устройства. Это позволяет использовать сигналы запроса UARTRXINTR и UARTRTXINTR для обеспечения чтения и записи данных согласованно с достижением заданного порога заполнения буферов FIFO приемника и передатчика.

Прерывание по обнаружению ошибки UARTRTEINTR формируется в случае возникновения той или иной ошибки приема данных. Предусмотрен ряд условий формирования признака ошибки.

Прерывание по состоянию модема представляет собой комбинацию признаков изменения отдельных линий состояния модема.

Прерывание UARTRNEINTR возникает в случае, если FIFO приемника получает хотя бы одно слово данных, то есть становится не пусто.

Прерывание UARTRTFEINTR выставляется, если FIFO передатчика не имеет никаких данных, то есть пусто в данный момент.

UARTTNBSYINTR выставляется в том случае, если сдвиговый регистр передатчика пуст.

Признаки возникновения каждого из условий прерывания можно считать либо из регистра прерываний UARTRIS, либо из маскированного регистра прерываний UARTMIS.

1.10.1 UARTMSINTR

Прерывание по состоянию модема возникает в случае изменения любой из линий состояний модема (nUARTCTS, nUARTDCD, nUARTDSR, nUARTRI). Сброс прерывания осуществляется путем записи 1 в соответствующий (в зависимости от линии состояния модема, вызвавшей прерывание) разряд регистра сброса прерывания ICR.

1.10.2 UARTRXINTR

Состояние прерывания от приемника может измениться в случае возникновения одного из следующих событий:

- буфер FIFO разрешён и его заполнение достигло заданного порогового значения. В этом случае линия прерывания переходит в высокое состояние. Сигнал прерывания переходит в низкое состояние после чтения данных из буфера приемника до тех пор, пока его заполнение не станет меньше порога, либо после сброса прерывания;
- буфер FIFO запрещен (имеет размер один символ), принят один символ данных. При этом линия прерывания переходит в высокое состояние. Сигнал прерывания переходит в низкое состояние после чтения одного байта данных, либо после сброса прерывания.

1.10.3 UARTTXINTR

Состояние прерывания от передатчика может измениться в случае возникновения одного из следующих событий:

- буфер FIFO разрешён и его заполнение меньше или равно заданному пороговому значению. В этом случае линия прерывания переходит в высокое состояние. Сигнал прерывания переходит в низкое состояние после записи данных в буфера передатчика до тех пор, пока его заполнение не станет больше порога, либо после сброса прерывания;
- буфер FIFO запрещен (имеет размер один символ), данные в буферном регистре передатчика отсутствуют. При этом линия прерывания переходит в высокое состояние. Сигнал прерывания переходит в низкое состояние после записи одного байта данных, либо после сброса прерывания.

Для занесения данных в буфер FIFO передатчика необходимо записать данные в буфер либо перед разрешением работы приемопередатчика и прерываний, либо после разрешения работы приемопередатчика и прерываний.

Примечание – Прерывание передатчика работает по фронту, а не по уровню сигнала. В случае если модуль и прерывания от него разрешены до осуществления записи данных в буфер FIFO передатчика, прерывание не формируется. Прерывание возникает только при опустошении буфера FIFO.

1.10.4 UARTRTINTR

Прерывание по таймауту приемника возникает в случае, если буфер FIFO приемника не пуст, и на вход приемника не поступало новых данных в течение периода времени, необходимого для передачи 32 бит. Прерывание по таймауту снимается либо после считывания данных из буфера приемника до его опустошения (или считывания одного байта в случае, если буфер FIFO запрещен), либо путем записи 1 в соответствующий бит регистра сброса прерывания ICR.

1.10.5 UARTEINTR

Прерывание по обнаружению ошибки происходит в случае возникновения ошибки при приеме данных. Оно может быть вызвано рядом факторов:

- ошибка в структуре кадра;
- ошибка контроля четности;
- разрыв линии;
- переполнение буфера.

Причину возникновения прерывания можно определить, прочитав содержимое регистра прерываний RIS, либо содержимое маскированного регистра прерываний MIS.

Сброс прерывания осуществляется путем записи соответствующих бит в регистр сброса прерывания ICR. За прерываниями по обнаружению ошибки закреплены биты с 7 по 10.

1.10.6 UARTINTR

Все описанные сигналы запроса на прерывание скомбинированы в общую линию путем объединения логическое ИЛИ сигналов UARTRXINTR, UARTRTXINTR, UARTRTINTR, UARTMSINTR, UARTEINTR, UARTRNEINTR, UARTRTFEINTR и UARTRTNBSYINTR с учетом маскирования. Общий выход может быть подключен к системному контроллеру прерывания, что позволит ввести дополнительное маскирование запросов на уровне периферийных устройств.

1.11 Программное управление модулем

Общая информация

Следующая информация применима ко всем регистрам контроллера:

- Базовый адрес контроллера фиксирован. Смещение каждого регистра относительно базового адреса постоянно.
- Не следует пытаться получить доступ к зарезервированным или неиспользуемым адресам. Это может привести к непредсказуемому поведению модуля.
- За исключением специально оговоренных в настоящей спецификации случаев:
 - не следует изменять значения не определенных в спецификации разрядов регистров;
 - не следует использовать значения не определенных в спецификации разрядов регистров;
 - все биты регистров (за исключением специально оговоренных случаев) устанавливаются в значение 0 после сброса по включению питания или системного сброса.
- Столбец «Тип» Таблица 367 определяет режим доступа к регистру в соответствии с обозначениями:
 - RW – чтение и запись;
 - RO – только чтение;
 - WO – только запись.

1.12 Обобщенные данные о регистрах устройства

Данные о регистрах модуля универсального асинхронного приемопередатчика приведены в таблице ниже.

Таблица 367 – Обобщенные данные о регистрах устройства

Базовый Адрес	Название				Описание
0x40008000	UART1				Регистры контроллера интерфейса UART1
0x40010000	UART2				Регистры контроллера интерфейса UART2
0x40038000	UART3				Регистры контроллера интерфейса UART3
0x400E8000	UART4				Регистры контроллера интерфейса UART4
Смещение	Наименование	Тип	Значение после сброса	Размер, бит	Описание
0x000	DR	RW	0x---	13/9	Регистр данных
0x004	RSR_ECR	RW	0x0	4/0	Регистра состояния приемника / Сброс ошибки приемника
0x008-0x014					Резерв
0x018	FR	RO	0b-10010---	9	Регистр флагов
0x01C					Резерв
0x020	ILPR	RW	0x00	8	Регистр управления ИК обменом в режиме пониженного энергопотребления
0x024	IBRD	RW	0x0000	16	Целая часть делителя скорости обмена данными
0x028	FBRD	RW	0x00	6	Дробная часть делителя скорости обмена данными
0x02C	LCR_H	RW	0x00	9	Регистр управления линией
0x030	CR	RW	0x0300	16	Регистр управления
0x034	IFLS	RW	0x12	6	Регистр порога прерывания по заполнению буфера FIFO
0x038	IMSC	RW	0x000	14	Регистр маски прерывания
0x03C	RIS	RO	0x300-	14	Регистр состояния прерываний
0x040	MIS	RO	0x00-	14	Регистр состояния прерываний с маскированием
0x044	ICR	WO	-	11	Регистр сброса прерывания
0x048	DMACR	RW	0x00	3	Регистр управления ПДП
0x080	TCR	RW	0x00	3	Регистр управления тестированием

1.12.1 Регистр данных DR

В ходе передачи данных:

Если буфер FIFO передатчика разрешен, то слово данных, записанное в рассматриваемый регистр, направляется в буфер FIFO передатчика.

В противном случае, записанное слово фиксируется в буферный регистр передатчика (последний элемент буфера FIFO).

Операция записи в регистр инициирует передачу данных. Слово данных предваряется стартовым битом, дополняется битом контроля четности (если режим контроля четности включен) и стоповым битом. Сформированное слово отправляется в линию передачи данных.

В ходе приема данных:

Если буфер FIFO приемника разрешен, байт данных и четыре бита состояния (разрыв, ошибка формирования кадра, четность, переполнение) сохраняются в 12-битном буфере.

В противном случае байт данных и биты состояния записываются в буферный регистр (последний элемент буфера FIFO).

Полученные из линии связи байты данных считываются путем чтения из регистра UART_DR принятых данных совместно с соответствующими битами состояния. Информация о состоянии также может быть получена путем чтения регистра RSR_ECR (Таблица 368).

Таблица 368 – Формат регистра UARTDR

Бит	Наименование	Назначение
15...13	-	Зарезервировано
12	OE	Переполнение буфера приемника. Бит устанавливается в 1 в случае, если на вход приемника поступают данные, в то время как буфер заполнен. Сбрасывается в 0 после того, как в буфере появится свободное место.
11	BE	Разрыв линии. Устанавливается в 1 при обнаружении признака разрыва линии, то есть в случае наличия низкого логического уровня на входе приемника в течение времени большего, чем длительность передачи полного слова данных (включая стартовый, стоповый биты и бит проверки на четность). При включенном FIFO данная ошибка ассоциируется с последним символом, поступившим в буфер. В случае обнаружения разрыва линии в буфер загружается только один нулевой символ, прием данных возобновляется только после перехода линии в логическую 1 и последующего обнаружения корректного стартового бита.
10	PE	Ошибка контроля четности. Устанавливается в 1 в случае, если четность принятого символа данных не соответствует установкам битов EPS и SPS в регистре управления линией LCR_H. При включенном FIFO данная ошибка ассоциируется с последним символом, поступившим в буфер.
9	FE	Ошибка в структуре кадра. Устанавливается в 1 в случае, если в принятом символе не обнаружен корректный стоповый бит (корректный стоповый бит равен 1). При включенном FIFO данная ошибка ассоциируется с последним символом, поступившим в буфер.
8...0	DATA	Принимаемые данные (чтение) Передаваемые данные (запись)

Примечание – Необходимо запрещать работу приемопередатчика перед любым перепрограммированием его регистров управления. Если приемопередатчик переводится в

отключенное состояние во время передачи или приема символа, то перед остановкой он завершает выполняемую операцию.

1.12.2 Регистр состояния приемника / сброса ошибки RSR_ECR

Состояние приемника также может быть считано из регистра RSR. В этом случае информация о состоянии признаков разрыва линии, ошибки контроля четности и ошибки в структуре кадра относится к последнему символу, считанному из регистра данных DR. С другой стороны, признак переполнения буфера устанавливается немедленно после возникновения этого состояния (и не связан с последним, считанным из регистра DR, байтом данных).

Запись в регистр ECR приводит к сбросу признаков ошибок переполнения, четности, структуры кадра, разрыва линии. Кроме того, все эти признаки устанавливаются в 0 после сброса устройства.

Таблица 369 показывает назначение бит регистра RSR_ECR.

Таблица 369 – Регистр RSR_ECR

Биты	Наименование	Назначение
7...4	-	Зарезервировано. При чтении результат не определен
3	OE	Переполнение буфера приемника. Бит устанавливается в 1 в случае, если на вход приемника поступают данные, в то время как буфер заполнен. Сбрасывается в 0 после записи в регистр ECR. Содержимое буфера остается верным, так как перезаписан был только регистр сдвига. Центральный процессор должен считать данные для того, чтобы освободить буфер FIFO.
2	BE	Разрыв линии. Устанавливается в 1 при обнаружении признака разрыва линии, то есть в случае наличия низкого логического уровня на входе приемника в течение времени, большего, чем длительность передачи полного слова данных (включая стартовый, стоповый биты и бит проверки на четность). Бит сбрасывается в 0 после записи в регистр ECR. При включенном FIFO данная ошибка ассоциируется с символом, находящемся на вершине буфера. В случае обнаружения разрыва линии в буфер загружается только один нулевой символ, прием данных возобновляется только после перехода линии в логическую 1 и последующего обнаружения корректного стартового бита.
1	PE	Ошибка контроля четности. Устанавливается в 1 в случае, если четность принятого символа данных не соответствует установкам битов EPS и SPS в регистре управления линией LCR_H. Бит сбрасывается в 0 после записи в регистр ECR. При включенном FIFO данная ошибка ассоциируется с символом, находящимся на вершине буфера.
0	FE	Ошибка в структуре кадра. Устанавливается в 1 в случае, если в принятом символе не обнаружен корректный стоповый бит (корректный стоповый бит равен 1). Бит сбрасывается в 0 после записи в регистр ECR. При включенном FIFO данная ошибка ассоциируется с символом, находящимся на вершине буфера.

Примечание – Перед чтением регистра состояния RSR необходимо считать данные, принятые из линии, путем обращения к регистру данных DR. Противоположная

последовательность действий не допускается, так как регистр RSR обновляет свое состояние только после чтения регистра DR. Вместе с тем, информация о состоянии приемника может быть получена непосредственно из регистра данных DR.

1.12.3 Регистр флагов FR

После сброса биты регистра флагов TXFF, RXFF и BUSY устанавливаются в 0, а биты TXFE и RXFE – в 1. Таблица 370 содержит информацию о назначении бит регистра.

Таблица 370 – Регистр FR

Биты	Наименование	Назначение
15...9		Резерв. Не модифицируйте. При чтении заполняются нулями.
8	RI	Инверсия линии nUARTRI
7	TXFE	Буфер FIFO передатчика пуст. Значение бита зависит от состояния бита FEN регистра управления линией LCR_H. Если буфер FIFO запрещен, бит устанавливается в 1, когда буферный регистр передатчика пуст. В противном случае он равен 1, если пуст буфер FIFO передатчика. Данный бит не дает никакой информации о наличии данных в регистре сдвига передатчика.
6	RXFF	Буфер FIFO приемника заполнен. Значение бита зависит от состояния бита FEN регистра управления линией LCR_H. Если буфер FIFO запрещен, бит устанавливается в 1, когда буферный регистр приемника занят. В противном случае он равен 1, если заполнен буфер FIFO приемника.
5	TXFF	Буфер FIFO передатчика заполнен. Значение бита зависит от состояния бита FEN регистра управления линией LCR_H. Если буфер FIFO запрещен, бит равен 1, когда буферный регистр передатчика занят. В противном случае он равен 1, если заполнен буфер FIFO передатчика.
4	RXFE	Буфер FIFO приемника пуст. Значение бита зависит от состояния бита FEN регистра управления линией LCR_H. Если буфер FIFO запрещен, бит устанавливается в 1, когда буферный регистр приемника пуст. В противном случае он равен 1, если пуст буфер FIFO приемника.
3	BUSY	UART занят. Бит устанавливается в 1 в случае, если контроллер передает в линию данные. Бит остается установленным до тех пор, пока данные, включая стоповые биты, не будут полностью переданы. Кроме того, бит занятости устанавливается в 1 при наличии данных в буфере FIFO передатчика, вне зависимости от состояния приемопередатчика (даже если он запрещен).
2	DCD	Инверсия линии nUARTDCD.
1	DSR	Инверсия линии nUARTDSR.
0	CTS	Инверсия линии nUARTCTS.

1.12.4 Регистр управления ИК обменом в режиме пониженного энергопотребления ILPR

Этот восьмиразрядный регистр, доступный для чтения и записи, содержит значение коэффициента деления частоты UART_CLK, для формирования тактового сигнала IrLPBaud16. Назначение разрядов регистра показано в таблице ниже (Таблица 371).

Требуемое значение коэффициента деления для формирования сигнала IrLPBaud16 вычисляется по формуле: $ILPDVSR = F_{UART_CLK} / F_{IrLPBaud16}$, где номинальное значение частоты $F_{IrLPBaud16}$ составляет 1,8432 МГц.

Коэффициент деления должен быть установлен таким образом, чтобы выполнялось соотношение: $1,42 \text{ МГц} < F_{IrLPBaud16} < 2,12 \text{ МГц}$, что, в свою очередь, гарантирует формирование кодеком импульсов данных с длительностью 1,41 – 2,11 мкс (в три раза длиннее периода сигнала IrLPBaud16).

Таблица 371 – Регистр ILPR

Биты	Наименование	Назначение
7...0	ILPDVSR	Коэффициент деления частоты UART_CLK, для формирования тактового сигнала IrLPBaud16. После сброса устанавливается в 0. Примечание – Коэффициент 0 – запрещенное значение. В случае его установки импульсы IrLPBaud16 формироваться не будут.

Примечание – В интересах подавления помех, при работе в режиме IrDA с пониженным энергопотреблением кодек игнорирует поступающие на вход SIRIN импульсы с длительностью, меньшей трех периодов сигнала IrLPBaud16.

1.12.5 Регистр целой части делителя скорости передачи данных IBRD

Назначение бит регистра представлено в таблице ниже.

Таблица 372 – Регистр IBRD

Биты	Наименование	Назначение
15...0	BAUDDIV_INT	Целая часть коэффициента деления частоты для формирования тактового сигнала передачи данных. После сброса устанавливается в 0.

1.12.6 Регистр дробной части делителя скорости передачи данных FBRD

Назначение бит регистра представлено в таблице ниже.

Таблица 373 – Регистр BFRD

Биты	Наименование	Назначение
5...0	BAUDDIV_FRAC	Дробная часть коэффициента деления частоты для формирования тактового сигнала передачи данных. После сброса устанавливается в 0.

Коэффициент деления вычисляется по формуле:

$$BAUDDIV = F_{UART_CLK} / (16 * \text{Baud_rate}),$$

где F_{UART_CLK} – тактовая частота контроллера UART, Baud_rate – требуемая скорость передачи данных (в бит/с).

Коэффициент BAUDDIV состоит из целой и дробной частей – BAUDDIV_INT и BAUDDIV_FRAC, соответственно.

Примечание – Изменение содержимого регистров IBRD и FBRD вступают в силу только после завершения передачи и приема текущего символа данных.

Минимальный допустимый коэффициент деления – 1, максимальный 65535 ($2^{16} - 1$). Таким образом, значение IBRD, равное 0 является недопустимым, при этом значение регистра FBRD игнорируется.

Аналогично, при IBRD равном 65535 (0xFFFF), значение FBRD не может быть больше нуля. Невыполнение этого условия приведет к прерыванию приема или передачи.

Далее приведен пример вычисления коэффициента деления.

Пример. Вычисление коэффициента деления

Пусть требуемая скорость передачи данных составляет 230400 бит/с, частота тактового сигнала $F_{UART_CLK} = 4$ МГц. Тогда:

Коэффициент деления = $(4 \cdot 10^6) / (16 \cdot 230400) = 1,085$.

Таким образом, BRDI = 1, BRDF = 0,085.

Следовательно, значение, записываемое в регистр BFRD, равно $m = \text{integer}((0.085 \cdot 64) + 0.5) = 5$.

Реальное значение коэффициента деления = $1 + 5/64 = 1.078$.

Реальная скорость передачи данных = $(4 \cdot 10^6) / (16 \cdot 1.078) = 231911$ бит/с.

Ошибка установки скорости = $(231911 - 230400) / 230400 \cdot 100\% = 0.656\%$.

Максимальная ошибка установки скорости передачи данных с использованием шестизрядного регистра BFRD = $1/64 \cdot 100\% = 1,56\%$. Такая ошибка возникает в случае $m = 1$, при этом разница накапливается в течение 64 тактовых интервалов.

Таблица 375 содержит значения коэффициента деления для типичных скоростей передачи данных при частоте $F_{UART_CLK} = 7,3728$ МГц. При таких параметрах дробная часть коэффициента деления не используется, следовательно, в регистр FBRD должен быть записан ноль.

Таблица 374 – Коэффициенты деления при частоте $F_{UART_CLK} = 7,3728$ МГц

Коэффициент деления	Скорость передачи данных
0x0001	460800
0x0002	230400
0x0004	115200
0x0006	76800
0x0008	57600
0x000C	38400
0x0018	19200
0x0020	14400
0x0030	9600
0x00C0	2400
0x0180	1200
0x105D	110

Таблица 375 содержит значения коэффициента деления для типичных скоростей передачи данных при частоте $F_{UART_CLK} = 4$ МГц.

Таблица 375 – Коэффициенты деления при частоте $F_{UART_CLK} = 4 \text{ МГц}$

Целая часть	Дробная часть	Требуемая скорость	Реальная скорость	Ошибка, %
0x001	0x05	230400	231911	0.656
0x002	0x0B	115200	115101	0.086
0x003	0x10	76800	76923	0.160
0x006	0x21	38400	38369	0.081
0x011	0x17	14400	14401	0.007
0x01A	0x03	9600	9598	0.021
0x068	0x0B	2400	2400	~0

1.12.7 Регистр управления линией LCR_H

Данный регистр обеспечивает доступ к разрядам с 29 по 22 регистра LCR. При сбросе все биты регистра LCR_H обнуляются.

Назначение разрядов регистра описано в таблице ниже.

Таблица 376 – Регистр LCR_H

Биты	Наименование	Назначение
15...9		Зарезервировано. Не модифицируйте. При чтении выдаются нули.
8	WLEN[2]	Длина слова – количество передаваемых или принимаемых информационных бит в кадре: 0 – длина слова определяется WLEN[1:0] 1 – 9 бит
7	SPS	Передача бита четности с фиксированным значением. 0 – запрещена; 1 – на месте бита четности передается инверсное значение бита EPS, оно же проверяется при приеме данных. (При EPS=0 на месте бита четности передается 1, при EPS=1 – передается 0). Значение бита SPS не играет роли в случае, если битом PEN формирование и проверка бита четности запрещено (Таблица 377)
6...5	WLEN[1:0]	Длина слова – количество передаваемых или принимаемых информационных бит в кадре: 0b11 – 8 бит, 0b10 – 7 бит, 0b01 – 6 бит, 0b00 – 5 бит
4	FEN	Разрешение работы буфера FIFO приемника и передатчика. 0 – запрещено, 1 – разрешено
3	STP2	Режим передачи двух стоповых бит. 0 – один стоповый бит; 1 – два стоповых бита. Приемник не проверяет наличие дополнительного стопового бита в кадре
2	EPS	Четность/нечетность. 0 – бит четности дополняет количество единиц в информационной части кадра до нечетного, 1 – до четного числа. Значение бита EPS не играет роли в случае, если битом PEN формирование и проверка бита четности запрещено (Таблица 377)
1	PEN	Разрешение проверки четности. 0 – кадр не содержит бита четности, 1 – бит четности передается в кадре и проверяется при приеме данных (Таблица 377)
0	BRK	Разрыв линии. Если этот бит установлен в 1, то по завершении передачи текущего символа на выходе линии UART_TXD устанавливается низкий уровень сигнала. Для

		правильного выполнения этой операции программное обеспечение должно обеспечить передачу сигнала разрыва в течение, как минимум, времени передачи двух информационных кадров. В нормальном режиме функционирования бит должен быть установлен в 0
--	--	--

Содержимое регистров LCR_H, IBRD и FBRD совместно образует общий 31-разрядный регистр LCR, который обновляется по стробу, формируемому при записи в LCR_H. Таким образом, для того, чтобы изменение параметров коэффициента деления частоты обмена данными вступило в силу, после их изменения значения регистров IBRD и/или FBRD необходимо осуществить запись данных в регистр LCR_H.

Таблица 377 содержит данные об истинности для бит управления контролем четности SPS, EPS, PEN регистра управления линией LCR_H.

Таблица 377 – Управление режимом контроля четности

PEN	EPS	SPS	Бит контроля четности
0	X	X	Не передается, не проверяется
1	1	0	Проверка четности слова данных
1	0	0	Проверка нечетности слова данных
1	0	1	Бит четности постоянно равен 1
1	1	1	Бит четности постоянно равен 0

Примечания:

Регистры LCR_H, IBRD и FBRD не должны изменяться:

При разрешенной работе приемопередатчика;

Во время завершения приема или передачи данных в процессе остановки (перевода в запрещенное состояние) приемопередатчика.

Целостность данных в буферах FIFO не гарантируется в следующих случаях:

После установки бита разрыва линии BRK;

Если программное обеспечение произвело остановку приемопередатчика при наличии данных в буферах FIFO, после его повторного перевода в разрешенное состояние.

1.12.8 Регистр управления CR

После сброса все биты регистра управления, за исключением битов 9 и 8 устанавливаются в нулевое состояние. Биты 9 и 8 устанавливаются в единичное состояние.

Назначение разрядов регистра управления показано в таблице ниже.

Таблица 378 – Регистр управления CR

Биты	Наименование	Назначение
15	CTSEn	Разрешение управления потоком данных по CTS. 1 – разрешено, данные передаются в линию только при активном значении сигнала nUARTCTS. 0 – запрещено.
14	RTSEn	Разрешение управления потоком данных по RTS. 1 – разрешено, запрос данных от внешнего устройства осуществляется только при наличии свободного места в буфере FIFO приемника. 0 – запрещено.

13	Out2	Инверсия сигнала на линии состояния модема nUARTOut2. В режиме окончного оборудования (DTE) эта линия может использоваться в качестве линии «сигнал вызова» (RI). 1 – сигнал разрешен. 0 – сигнал запрещен.
12	Out1	Инверсия сигнала на линии состояния модема nUARTOut1. В режиме окончного оборудования (DTE) эта линия может использоваться в качестве линии «обнаружен информационный сигнал» (DCD). 1 – сигнал разрешен. 0 – сигнал запрещен.
11	RTS	Инверсия сигнала на линии состояния модема nUARTRTS. 1 – сигнал разрешен. 0 – сигнал запрещен.
10	DTR	Инверсия сигнала на линии состояния модема nUARTDTR. 1 – сигнал разрешен. 0 – сигнал запрещен.
9	RXE	Разрешение приема. 1 – работа приемника разрешена. Прием данных осуществляется либо по интерфейсу асинхронного последовательного обмена, либо по интерфейсу ИК обмена SIR, в зависимости от значения бита SIREN. В случае перевода приемопередатчика в запрещенное состояние в ходе приема данных, он завершает прием текущего символа перед остановкой. 0 – работа приемника запрещена.
8	TXE	Разрешение передачи. 1 – работа передатчика разрешена. Передача осуществляется либо по интерфейсу асинхронного последовательного обмена, либо по интерфейсу ИК обмена SIR, в зависимости от значения бита SIREN. В случае перевода приемопередатчик в запрещенное состояние в ходе передачи данных, он завершает передачу текущего символа перед остановкой. 0 – работа передатчика запрещена.
7	LBE	Режим тестирования по шлейфу 1 – шлейф разрешен. 0 – запрещен. В режиме разрешенного шлейфа: Если установлены бит SIREN=1 и бит регистра управления тестированием TCR SIRTEST=1, то сигнал с выхода кодека nSIROUT инвертируется и подается на вход кодека SIRIN. Бит SIRTEST устанавливается в 1 для того, чтобы вывести устройство из полудуплексного режима, характерного для интерфейса SIR. После окончания тестирования по шлейфу бит SIRTEST должен быть установлен в 0. Если бит SIRTEST=0, то выходная линия передатчика UART_TXDx коммутируется на вход приемника UART_RXDx. Как в режиме SIR, так и в режиме UART, выходные линии состояния модема коммутируются на соответствующие входные линии. После сброса бит устанавливается в 0.
6...3		Резерв. Не модифицируйте. При чтении выдаются нули.

2	SIRLP	Выбор режима ИК обмена с пониженным энергопотреблением. 1 – длительность импульсов данных равна трем тактам сигнала IrLPBaud16 вне зависимости от выбранной скорости передачи данных. Выбор этого режима снижает энергопотребление, однако может привести к уменьшению дальности связи. 0 – длительность импульсов данных равна 3/16 длительности передачи бита.
1	SIREN	Разрешение работы кодека ИК передачи данных IrDA SIR: 1 – разрешена работа кодека ИК. Данные передаются на выход nSIROUT и принимаются с входа SIRIN. Линия UART_TXDx находится в высоком состоянии. Данные на входе UART_RXDx и линиях состояния модема не обрабатываются. В случае если UARTEN=0 значение бита не играет роли. 0 – запрещен. Сигнал nSIROUT находится в низком состоянии, данные на входе SIRIN не обрабатываются.
0	UARTEN	Разрешение работы приемопередатчика. 0 – работа запрещена. Перед остановкой завершается прием и/или передача обрабатываемого в текущий момент символа. 1 – работа разрешена. Производится обмен данными либо по линиям асинхронного обмена, либо по линиям ИК обмена SIR, в зависимости от состояния бита SIREN.

Примечание – Для того чтобы разрешить передачу данных, необходимо установить в 1 биты TXE и UARTEN. Аналогично, для разрешения приема данных необходимо установить в 1 биты RXE и UARTEN.

Примечание – Рекомендуются следующая последовательность действий для программирования регистров управления:

Остановите работу приемопередатчика.

Дождитесь окончания приема и/или передачи текущего символа данных.

Сбросьте буфер передатчика путем установки бита FEN регистра LCR_H в 0.

Изменить настройки регистра CR.

Возобновите работу приемопередатчика.

1.12.9 Регистр порога прерывания по заполнению буфера FIFO IFLS

Данный регистр используется для установки порогового значения заполнения буферов передатчика и приемника, по достижению которых генерируется сигнал прерывания UARTTXINTR или UARTRXINTR, соответственно. Прерывание генерируется в момент перехода величины заполнения буфера через заданное значение.

После сброса в регистре устанавливается порог, соответствующий заполнению половины буфера. Формат регистра и значения его битов представлены в таблице ниже.

Таблица 379 – Регистр IFLS

Биты	Наименование	Назначение
15...6		Резерв. Не модифицируйте. При чтении выдаются нули.
5...3	RXIFLSEL	Порог прерывания по заполнению буфера приемника: b000 = Буфер заполнен на 1/8 b001 = Буфер заполнен на 1/4 b010 = Буфер заполнен на 1/2 b011 = Буфер заполнен на 3/4

		b100 = Буфер заполнен на 7/8 b101-b111 = резерв.
2...0	TXIFLSEL	Порог прерывания по заполнению буфера передатчика: b000 = Буфер заполнен на 1/8 b001 = Буфер заполнен на 1/4 b010 = Буфер заполнен на 1/2 b011 = Буфер заполнен на 3/4 b100 = Буфер заполнен на 7/8 b101-b111 = резерв. Также стоит помнить, что в случае, когда сдвиговый регистр передатчика пуст, то слово, записанное в FIFO, будет сразу же переписано в сдвиговый регистр. Следовательно, для генерирования события прерывания от передатчика блока UART необходимо произвести запись в FIFO такого количества слов, которое превысит установленный порог хотя бы на одно слово с учетом описанного случая.

1.12.10 Регистр установки сброса маски прерывания IMSC

При чтении выдается текущее значение маски. При записи производится установка или сброс маски на соответствующее прерывание.

После сброса все биты регистра маски устанавливаются в нулевое состояние.

Назначение битов регистра IMSC показано в таблице ниже.

Таблица 380 – Регистр IMSC

Биты	Наименование	Назначение
31...14		Резерв. Не модифицируйте. При чтении выдаются нули.
13	TNBSYIM	Маска прерывания при отсутствии данных в сдвиговом регистре передатчика UARTTNBSYINTR: 1 – установлена; 0 – сброшена
12	TFEIM	Маска прерывания по отсутствию данных в FIFO передатчика UARTTFEINTR: 1 – установлена; 0 – сброшена
11	RNEIM	Маска прерывания по наличию данных в FIFO приемника UARTRNEINTR: 1 – установлена; 0 – сброшена
10	OEIM	Маска прерывания по переполнению буфера UARTOEINTR. 1 – установлена; 0 – сброшена.
9	BEIM	Маска прерывания по разрыву линии UARTBEINTR. 1 – установлена; 0 – сброшена.
8	PEIM	Маска прерывания по ошибке контроля четности UARTPEINTR. 1 – установлена; 0 – сброшена.
7	FEIM	Маска прерывания по ошибке в структуре кадра UARTFEINTR. 1 – установлена; 0 – сброшена.
6	RTIM	Маска прерывания по таймауту приема данных UARTRTINTR. 1 – установлена; 0 – сброшена.
5	TXIM	Маска прерывания от передатчика UARTTXINTR. 1 – установлена; 0 – сброшена.
4	RXIM	Маска прерывания от приемника UARTRXINTR. 1 – установлена; 0 – сброшена.
3	DSRMIM	Маска прерывания UARTDSRINTR по изменению состояния линии nUARTDSR. 1 – установлена; 0 – сброшена.
2	DCDMIM	Маска прерывания UARTDCDINTR по изменению состояния линии nUARTDCD. 1 – установлена; 0 – сброшена.

1	CTSMIM	Маска прерывания UARTCTSINTR по изменению состояния линии nUARTCTS. 1 – установлена; 0 – сброшена.
0	RIMIM	Маска прерывания UARTRIINTR по изменению состояния линии nUARTRI. 1 – установлена; 0 – сброшена.

ПРЕДВАРИТЕЛЬНО

1.12.11 Регистр состояния прерываний RIS

Этот регистр доступен только для чтения и содержит текущее состояние прерываний без учета маскирования. Данные, записываемые в регистр, игнорируются.

Предупреждение. После сброса все биты регистра, за исключением битов прерывания по состоянию модема (биты с 3 по 0), устанавливаются в 0. Значение битов прерывания по состоянию модема после сброса не определено.

Назначение бит в регистре RIS представлено в таблице ниже.

Таблица 381 – Регистр RIS

Биты	Наименование	Назначение
31...14		Резерв. Не модифицируйте. При чтении выдаются нули.
13	TNBSYRIS	Состояние прерывания при отсутствии данных в сдвиговом регистре передатчика UARTTNBSYINTR
12	TFERIS	Состояние прерывания по отсутствию данных в FIFO передатчика UARTTFEINTR
11	RNERIS	Состояние прерывания по наличию данных в FIFO приемника UARTRNEINTR
10	OERIS	Состояние прерывания по переполнению буфера UARTOEINTR. 1 – буфер приемника переполнен; 0 – буфер приемника не переполнен.
9	BERIS	Состояние прерывания по разрыву линии UARTBEINTR. 1 – произошел разрыв линии приема; 0 – разрыва линии приема не происходило.
8	PERIS	Состояние прерывания по ошибке контроля четности UARTPEINTR. 1 – возникла ошибка контроля четности; 0 – ошибки контроля четности не возникало.
7	FERIS	Состояние прерывания по ошибке в структуре кадра UARTFEINTR. 1 – возникла ошибка в структуре кадра; 0 – ошибки в структуре кадра не возникало.
6	RTRIS	Состояние прерывания по таймауту приема данных UARTRTINTR ¹ . 1 – вышло время таймаута приема данных; 0 – время таймаута приема данных не вышло.
5	TXRIS	Состояние прерывания от передатчика UARTTXINTR. 1 – возникло прерывание от передатчика; 0 – прерывания от передатчика нет.
4	RXRIS	Состояние прерывания от приемника UARTRXINTR. 1 – возникло прерывание от приемника; 0 – прерывание от приемника не возникало.
3	DSRRMIS	Состояние прерывания UARTDSRINTR по изменению линии nUARTDSR. 1 – возникло прерывание; 0 – прерывание не возникало.
2	DCDRMIS	Состояние прерывания UARTDCDINTR по изменению линии nUARTDCD. 1 – возникло прерывание; 0 – прерывание не возникало.

¹ Сигнал маски прерывания по таймауту используется в качестве разрешения перехода в режим пониженного энергопотребления. Поэтому чтение состояния прерывания по таймауту из регистров MIS и RIS даст одинаковый результат.

1	CTSRMIS	Состояние прерывания UARTCTSINTR по изменению линии nUARTCTS. 1 – возникло прерывание; 0 – прерывание не возникало.
0	RIRMIS	Состояние прерывания UARTRIINTR по изменению линии nUARTRI. 1 – возникло прерывание; 0 – прерывание не возникало.

1.12.12 Регистр маскированного состояния прерываний MIS

Этот регистр доступен только для чтения и содержит текущее состояние прерываний с учетом маскирования. Данные, записываемые в регистр, игнорируются.

После сброса все биты регистра, за исключением битов прерывания по состоянию модема (биты с 3 по 0), устанавливаются в 0. Значение битов прерывания по состоянию модема после сброса не определено.

Назначение бит в регистре MIS представлено в таблице ниже.

Таблица 382 – Регистр MIS

Биты	Наименование	Назначение
31...14		Резерв. Не модифицируйте. При чтении выдаются нули.
13	TNBSYMIS	Маскированное состояние прерывания при отсутствии данных в сдвиговом регистре передатчика UARTTNBSYINTR
12	TFEMIS	Маскированное состояние прерывания по отсутствию данных в FIFO передатчика UARTTFEINTR
11	RNEMIS	Маскированное состояние прерывания по наличию данных в FIFO приемника UARTRNEINTR
10	OEMIS	Маскированное состояние прерывания по переполнению буфера UARTOEINTR. 1 – буфер приемника переполнен; 0 – буфер приемника не переполнен.
9	BEMIS	Маскированное состояние прерывания по разрыву линии UARTBEINTR. 1 – произошел разрыв линии приема; 0 – разрыва линии приема не происходило.
8	PEMIS	Маскированное состояние прерывания по ошибке контроля четности UARTPEINTR. 1 – возникла ошибка контроля четности; 0 – ошибки контроля четности не возникало.
7	FEMIS	Маскированное состояние прерывания по ошибке в структуре кадра UARTFEINTR. 1 – возникла ошибка в структуре кадра; 0 – ошибки в структуре кадра не возникало.
6	RTMIS	Маскированное состояние прерывания по таймауту приема данных UARTRTINTR. 1 – вышло время таймаута приема данных; 0 – время таймаута приема данных не вышло.
5	TXMIS	Маскированное состояние прерывания от передатчика UARTTXINTR. 1 – возникло прерывание от передатчика; 0 – прерывания от передатчика нет.
4	RXMIS	Маскированное состояние прерывания от приемника UARTRXINTR. 1 – возникло прерывание от приемника; 0 – прерывание от приемника не возникало.

3	DSRMMIS	Маскированное состояние прерывания UARTDSRINTR по изменению линии nUARTDSR. 1 – возникло прерывание; 0 – прерывание не возникало.
2	DCDMMIS	Маскированное состояние прерывания UARTDCDINTR по изменению линии nUARTDCD. 1 – возникло прерывание; 0 – прерывание не возникало.
1	CTSMMIS	Маскированное состояние прерывания UARTCTSINTR по изменению линии nUARTCTS. 1 – возникло прерывание; 0 – прерывание не возникало.
0	RIMMIS	Маскированное состояние прерывания UARTRIINTR по изменению линии nUARTRI. 1 – возникло прерывание; 0 – прерывание не возникало.

1.12.13 Регистр сброса прерываний ICR

Этот регистр доступен только для записи и предназначен для сброса признака прерывания по заданному событию путем записи 1 в соответствующий бит. Запись 0 в любой из разрядов регистра игнорируется.

Назначение бит в регистре ICR представлено в таблице ниже.

Таблица 383 – Регистр ICR

Биты	Наименование	Назначение
31...11		Резерв. Не модифицируйте. При чтении выдаются нули.
10	OEIC	Сброс прерывания по переполнению буфера UARTOEINTR. 1 – сброс прерывания; 0 – не влияет на состояние регистра.
9	BEIC	Сброс прерывания по разрыву линии UARTBEINTR. 1 – сброс прерывания; 0 – не влияет на состояние регистра.
8	PEIC	Сброс прерывания по ошибке контроля четности UARTPEINTR. 1 – сброс прерывания; 0 – не влияет на состояние регистра.
7	FEIC	Сброс прерывания по ошибке в структуре кадра UARTFEINTR. 1 – сброс прерывания; 0 – не влияет на состояние регистра.
6	RTIC	Сброс прерывания по таймауту приема данных UARTRTINTR. 1 – сброс прерывания; 0 – не влияет на состояние регистра.
5	TXIC	Сброс прерывания от передатчика UARCTXINTR. 1 – сброс прерывания; 0 – не влияет на состояние регистра.
4	RXIC	Сброс прерывания от приемника UARTRXINTR. 1 – сброс прерывания; 0 – не влияет на состояние регистра.
3	DSRMIC	Сброс прерывания UARTDSRINTR по изменению линии nUARTDSR. 1 – сброс прерывания;

		0 – не влияет на состояние регистра.
2	DCDMIC	Сброс прерывания UARTDCDINTR по изменению линии nUARTDCD. 1 – сброс прерывания; 0 – не влияет на состояние регистра.
1	CTSMIC	Сброс прерывания UARTCTSINTR по изменению линии nUARTCTS. 1 – сброс прерывания; 0 – не влияет на состояние регистра.
0	RIMIC	Сброс прерывания UARTRIINTR по изменению линии nUARTRI. 1 – сброс прерывания; 0 – не влияет на состояние регистра.

ПРЕДВАРИТЕЛЬНО

1.12.14 Регистр управления прямым доступом к памяти DMACR

Регистр доступен по чтению и записи. После сброса все биты регистра обнуляются. Назначение бит регистра DMACR представлено в таблице ниже.

Таблица 384 – Регистр DMACR

Биты	Наименование	Назначение
31...3		Резерв. Не модифицируйте. При чтении выдаются нули.
2	DMAONERR	Блокирование DMA запросов при приеме при возникновении ошибок на линии. 1 – в случае возникновения прерывания по обнаружению ошибки блокируются запросы DMA от приемника UARTRXDMASREQ и UARTRXDMABREQ. 0 – DMA запросы не блокируются.
1	TXDMAE	Использование DMA при передаче. 1 – разрешено формирование запросов DMA для обслуживания буфера FIFO передатчика. 0 – запрещено формирование DMA запросов.
0	RXDMAE	Использование DMA при приеме. 1 – разрешено формирование запросов DMA для обслуживания буфера FIFO приемника; 0 – запрещено формирование DMA запросов.

1.12.15 Регистр управления тестированием TCR

Регистр доступен по чтению и записи. После сброса все биты регистра обнуляются. Назначение бит регистра TCR представлено в таблице ниже.

Таблица 385 – Регистр TCR

Биты	Наименование	Назначение
15...13		Резерв. При чтении значение непредсказуемо.
12...3	-	Зарезервировано
2	SIRSTEST	Разрешение приёма данных в кольцевом режиме с выхода IrDA передатчика. 1 – разрешено 0 – запрещено Используется совместно с установкой бита LBE в регистре CR
1	TESTFIFO	Разрешение чтения данных из FIFO передатчика и запись в FIFO приёмника. 1 – разрешено 0 – запрещено
0	ITEN	Перевод контроллера UART в тестовый режим 1 – тестовый режим разрешён 0 – тестовый режим запрещён

Контроллер прямого доступа в память DMA

1.1 Основные свойства контроллера DMA

Основные свойства и отличительные особенности:

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

1.2 Термины и определения

Таблица 386 – Термины и определения

Альтернативная	Альтернативная структура управляющих данных канала. Вы можете установить соответствующий регистр для изменения типа структуры данных (см. раздел «Структура управляющих данных канала»)
С	Идентификатор номера канала прямого доступа. Например: C=1 – канал DMA 1 C=23 – канал DMA 23
Канал	Возможны конфигурации контроллера с числом каналов до 32. Каждый канал содержит независимые сигналы управления передачей данных, которые могут инициировать передачу данных по каналу DMA
Управляющие данные канала	Структура данных находится в системной памяти. Вы можете программировать эту структуру данных так, что контроллер может выполнять передачу данных по каналу DMA в желаемом режиме. Контроллер должен иметь доступ к области системной памяти, где находится эта информация. Примечание – Любое упоминание в спецификации структуры данных означает управляющие данные канала
Цикл DMA	Все передачи DMA, которые контроллер должен выполнить для передачи N пакетов данных

Передача DMA	Акция пересылки одного байта, полуслова или слова.
N	Общее количество передач DMA, которые контроллер выполняет для канала
Пинг-понг	Режим работы для выбранного канала, при котором контроллер получает начальный запрос и затем выполняет цикл DMA, используя первичную или альтернативную структуру данных. После завершения этого цикла DMA контроллер начинает выполнять новый цикл DMA, используя другую (первичную или альтернативную) структуру данных. Контроллер сигнализирует об окончании каждого цикла DMA, позволяя главному процессору перенастраивать неактивную структуру данных. Контроллер продолжает переключаться от первичной к альтернативной структуре данных и обратно до тех пор, пока он не прочитает «неправильную» структуру данных, или пока он не завершит цикл без переключения к другой структуре
Первичная	Первичная структура управляющих данных канала. Контроллер использует эту структуру данных, если соответствующий разряд в регистре <code>chnl_pri_alt_set</code> установлен в 0.
R	Степень числа 2, устанавливающее число передач DMA, которые могут произойти перед сменой арбитража. Количество передач DMA программируется в диапазоне от 1 до 1024 двоичными шагами от 2 в степени 0 до 2 в степени 10
Исполнение с изменением конфигурации	Режим работы для выбранного канала, при котором контроллер получает запрос от периферии и выполняет 4 DMA передачи, используя первичную структуру управляющих данных, которые настраивают альтернативную структуру управляющих данных. После чего контроллер начинает цикл DMA, используя альтернативную структуру данных. После того, как цикл закончится и, если периферия устанавливает новый запрос на обслуживание, контроллер выполняет снова 4 DMA передачи, используя первичную структуру управляющих данных, которые опять перенастраивают альтернативную структуру управляющих данных. После чего контроллер начинает цикл DMA, используя альтернативную структуру данных. Контроллер будет продолжать работать вышеописанным способом до тех пор, пока не прочитает неправильную структуру данных или процессор не установит альтернативную структуру данных для обычного цикла. Контроллер устанавливает флаг <code>dma_done</code> , если окончание подобного режима работы происходит после выполнения обычного цикла.

1.3 Функциональное описание

На рисунке показана упрощенная структурная схема контроллера.

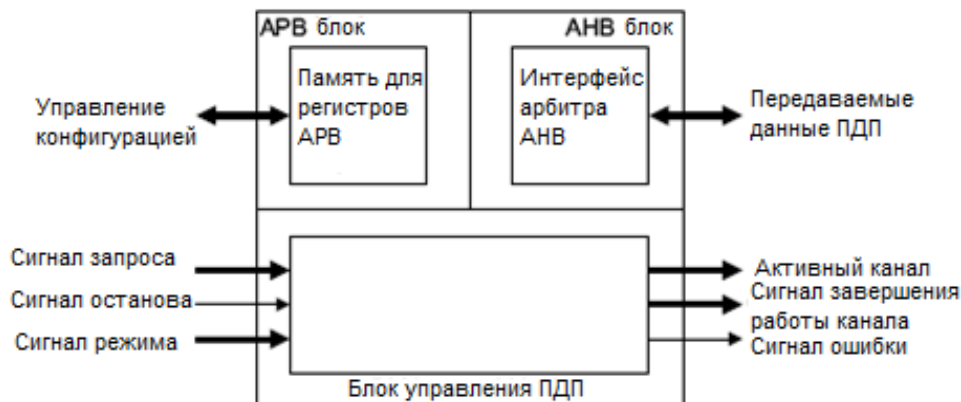


Рисунок 82 – Структурная схема контроллера

Контроллер состоит из следующих основных функциональных блоков:

- блок, подключенный к шине APB;
- блок, подключенный к шине AHB;
- управляющий блок DMA.

1.3.1 Распределение каналов DMA

Таблица 387 – Распределение каналов DMA

Номер запроса	dma_req	dma_sreq	Описание
0	UART1TXDMABREQ	UART1TXDMASREQ	Запрос от передатчика UART1
1	UART1RXDMABREQ	UART1RXDMASREQ	Запрос от приёмника UART1
2	UART2TXDMABREQ	UART2TXDMASREQ	Запрос от передатчика UART2
3	UART2RXDMABREQ	UART2RXDMASREQ	Запрос от приёмника UART2
4	SSP1TXDMABREQ	SSP1TXDMASREQ	Запрос от передатчика SSP1
5	SSP1RXDMABREQ	SSP1RXDMASREQ	Запрос от приёмника SSP1
6	CRCDMABREQ	CRCDMASREQ	Запрос от аппаратного блока вычисления CRC
7	UART3TXDMABREQ	UART3TXDMASREQ	Запрос от передатчика UART3
8	UART3RXDMABREQ	UART3RXDMASREQ	Запрос от приёмника UART3
9	TIM3DMAREQ	TIM3DMAREQ	Запрос от таймера общего назначения TIMER3
10	TIM1DMAREQ	TIM1DMAREQ	Запрос от таймера общего назначения TIMER1
11	TIM2DMAREQ	TIM2DMAREQ	Запрос от таймера общего назначения TIMER2
12	ADCIUDMABREQ1	ADCIUDMASREQ1	Запросы от блока АЦП для измерения напряжений и токов в электрической сети
13	ADCIUDMABREQ2	ADCIUDMASREQ2	
14	ADCIUDMABREQ3	ADCIUDMASREQ3	

15	ADCIUDMABREQ4	ADCIUDMASREQ4	
16	ADCIUDMABREQ5	ADCIUDMASREQ5	
17	ADCIUDMABREQ6	ADCIUDMASREQ6	
18	ADCIUDMABREQ7	ADCIUDMASREQ7	
19	SSP2TXDMABREQ	SSP2TXDMASREQ	Запрос от передатчика SSP2
20	SSP2RXDMABREQ	SSP2RXDMASREQ	Запрос от приёмника SSP2
21	SSP3TXDMABREQ	SSP3TXDMASREQ	Запрос от передатчика SSP3
22	SSP3RXDMABREQ	SSP3RXDMASREQ	Запрос от приёмника SSP3
23	TIM4DMAREQ	TIM4DMAREQ	Запрос от таймера общего назначения TIMER4
24	UART4TXDMABREQ	UART4TXDMASREQ	Запрос от передатчика UART4
25	UART4RXDMABREQ	UART4RXDMASREQ	Запрос от приёмника UART4
26	-	-	-
27	-	-	-
28	-	-	-
29	-	-	-
30	-	ADCDMASREQ	Запрос от АЦП последовательных приближений
31	-	-	-

1.3.2 Блок, подключенный к шине APB

Блок содержит набор регистров, позволяющих настраивать контроллер, используя ведомый APB интерфейс. Регистры занимают адресное пространство емкостью 4 Кбайт.

1.3.3 Блок, подключенный к шине AHB

Контроллер содержит один блок типа «ведущий» шины DMA Bus, который позволяет, используя 32-разрядную шину, передавать данные от источника к приемнику. Источник и приемник являются ведомыми шины AHB.

1.3.4 Управляющий блок DMA

Этот блок содержит схему управления, позволяющую реализовать следующие функции:

- осуществление арбитража поступающих запросов;
- индикацию активного канала;
- индикацию завершения обмена по каналу;
- индикацию состояния ошибки обмена по шине DMA Bus;
- разрешение медленным устройствам приостанавливать исполнение цикла DMA;
- ожидание запроса на очистку до завершения цикла DMA;
- осуществление одиночных или множественных передач DMA для каждого запроса;
- осуществление следующих типов DMA передач:
 - память – память;
 - память – периферия;
 - периферия – память.

ПРЕДВАРИТЕЛЬНО

1.3.5 Типы передач

Контроллер интерфейса не поддерживает пакетные передачи. Контроллер выполняет одиночные передачи. Отсутствие возможности осуществлять пакетные передачи оказывает минимальное влияние на производительность системы, так как пакетные передачи более эффективны в одноуровневых системах с шиной АНВ, где блоки должны «захватывать» шину или обращаться к внешней памяти. В тоже время контроллер DMA предназначен для использования в многоуровневых системах с шиной АНВ, включающих встроенную память.

1.3.6 Разрядность передаваемых данных

Контроллер интерфейса предоставляет возможность осуществлять передачу 8, 16 и 32 разрядных данных. Таблица 374 отображает значения комбинаций шины HSIZE.

Таблица 388 – Комбинации шины HSIZE

HSIZE[2]*	HSIZE[1]	HSIZE[0]	Разрядность данных (бит)
0	0	0	8
0	0	1	16
0	1	0	32
	1	1	**

* - сигнал постоянно удерживается в состоянии логический ноль.

** - запрещенная комбинация

Контроллер всегда использует передачи 32-разрядными данными при обращении к управляющим данным канала. Необходимо устанавливать разрядность данных источника соответствующую разрядности данных приемника.

1.3.7 Управление защитой данных

Контроллер позволяет устанавливать режимы защиты данных протокола АНВ-Lite, определяемые шиной HPROT[3:1]. Возможен выбор следующих режимов защиты:

- кэширование;
- буферизация;
- привилегированный.

Таблица 389 перечисляет значения комбинаций шины HPROT.

Таблица 389 – Режимы защиты данных

HPROT[3] кэширование	HPROT[2] буферизация	HPROT[1] привилегиро- ванный	HPROT[0] данные/команда	Описание
-	-	-	1*	Доступ к данным
-	-	0	-	Пользовательский доступ
-	-	1	-	Привилегированный доступ
-	0	-	-	Без буферизации
-	1	-	-	Буферизованный
0	-	-	-	Без кэширования
1	-	-	-	Кэшированный

Контроллер удерживает HPROT[0] в состоянии логической единицы, чтобы обозначить доступ к данным.

Для каждого цикла DMA возможен выбор режимов защиты данных передач источника и приемника. Более подробно это описано в разделе «Настройка управляющих данных».

Для каждого канала DMA также возможен выбор режима защиты данных. Более подробно это описано в разделе Управление DMA.

1.3.8 Инкремент адреса

Контроллер позволяет управлять инкрементом адреса при чтении данных из источника и при записи данных в приемник. Инкремент адреса зависит от разрядности передаваемых данных. В следующей таблице перечислены возможные комбинации.

Таблица 390 – Инкремент адреса

Разрядность данных	Величина инкремента
8	Байт, полуслово, слово
16	Полуслово, слово
32	Слово

Минимальная величина инкремента адреса всегда соответствует разрядности передаваемых данных. Максимальная величина инкремента адреса, осуществляемая контроллером, одно слово. Более подробно о настройке инкремента адреса написано в разделе Настройка управляющих данных. Этот раздел описывает разряды управления величиной инкремента адреса в управляющих данных канала.

Примечание – Если необходимо оставлять адрес неизменным при чтении или записи данных, для примера, при работе с FIFO, можно соответствующим образом настроить контроллер на работу с фиксированным адресом (см. раздел «Структура управляющих данных канала»).

1.4 Управление DMA

1.4.1 Правила обмена данными

Контроллер использует правила обмена данными (Таблица 391), при соблюдении следующих условий:

- канал DMA включен, что выполняется установкой в состояние логической единицы разрядов управления `chnl_enable_set[C]` и `master_enable`;
- флаги запроса `dma_req[C]` и `dma_sreq[C]` не замаскированы, что выполняется установкой в состояние логического нуля разряда управления `chnl_req_mask_set [C]`;
- контроллер находится не в тестовом режиме, что выполняется установкой в состояние логического нуля разряда управления `int_test_en_bit[C]`.

Таблица 391 – Правила, при которых передача данных по каналам разрешена, и запросы не маскируются

Правило	Описание
---------	----------

Правило	Описание
1	Если dma_active[C] установлен в 0, то установка в 1 dma_req[C] или dma_sreq[C] на один или более тактов сигнала HCLK, следующих или не следующих друг за другом, инициирует передачу по каналу номер C
2	Контроллер осуществляет установку в 1 только одного разряда dma_active[C]
3	Контроллер устанавливает в 1 dma_active[C] в момент начала передачи по каналу C
4	Для типов циклов DMA, отличных от периферийного «Исполнение с изменением конфигурации», dma_active[C] остается в состоянии 1 до тех пор, пока контроллер не окончит передачи с номерами меньше, чем значение 2 ^R или чем число передач, указанное в регистре n_minus_1. В периферийном режиме «Исполнение с изменением конфигурации», dma_active[C] остается в состоянии 1 в течение каждой пары DMA передач, с использованием первичной и альтернативной структур управляющих данных. Таким образом, контроллер выполняет 2 ^R передач, используя первичную структуру управляющих данных, затем без осуществления арбитража выполняет передачи с номерами меньше, чем значение 2 ^R (или чем число передач, указанное в регистре n_minus_1), используя альтернативную структуру управляющих данных. По окончании последней передачи dma_active[C] сбрасывается в 0
5	Контроллер устанавливает dma_active[C] в 0 как минимум на один такт сигнала HCLK перед тем, как снова установит dma_active[C] или dma_active[] в 1
6	Для каналов, по которым разрешена передача, контроллер осуществляет установку в 1 только одного dma_done[]
7	Если dma_req[C] устанавливается в состояние 1 в момент, когда dma_active[C] или dma_stall также в состоянии 1, то это означает, что контроллер обнаружил запрос
8	Если разряды cycle_ctrl для канала установлены в состояние 3'b100, 3'b101, 3'b110, 3'b111, то dma_done[C] никогда не будет установлен в 1
9	Если все передачи по каналу завершены, и разряды cycle_ctrl позволяют удержание dma_done[C], то по срезу сигнала dma_active[] произойдут события: – если dma_stall в состоянии 0, контроллер устанавливает dma_done[] в состояние 1 продолжительностью один такт HCLK – если dma_stall в состоянии 1, работа контроллера приостановлена. После того, как dma_stall будет установлен в 0, контроллер устанавливает dma_done[] в состояние 1 продолжительностью один такт HCLK
10	Состояние dma_waitonreq[C] можно изменять только при выключенном канале
11	Если dma_waitonreq[C] в состоянии 1, то сигнал dma_active[C] не перейдет в состояние 0 до тех пор, пока: – контроллер завершит 2 ^R передач (или число передач, указанное в регистре n_minus_1); – dma_req[C] будет установлен в 0; – dma_sreq[C] будет установлен в 0
12	Если за один такт сигнала HCLK перед установкой dma_active[C] в 0 dma_stall устанавливается в 1, то – контроллер установит dma_active[C] в 0 на следующем такте сигнала HCLK; – передача по каналу C не завершится, пока не будет сброшен в 0 dma_stall
13	Контроллер игнорирует dma_sreq[C], если dma_waitonreq[C] в состоянии 0
14	Контроллер игнорирует dma_sreq[C], если chnl_useburst_set[C] в состоянии 1 ^{*)}
15	Для циклов DMA, отличных по типу от периферийного режима «Исполнение с изменением конфигурации», по окончании 2 ^R передач контроллер устанавливает значение chnl_useburst_set[C] в состояние 0, если количество оставшихся передач меньше, чем 2 ^R . В периферийном режиме «Исполнение с изменением конфигурации» контроллер устанавливает значение chnl_useburst_set[C] в состояние 0 только, если количество оставшихся передач с использованием альтернативной структуры управляющих данных меньше, чем 2 ^R .

Правило	Описание
16	Для типов циклов DMA, отличных от периферийного режима «Исполнение с изменением конфигурации», если за один такт HCLK до установки dma_active[C] в 1 dma_sreq[C] и dma_waitonreq[C] установлены в 1 и dma_req[C] установлен в 0, то контроллер выполняет одну DMA передачу. В периферийном режиме «Исполнение с изменением конфигурации», если за один такт HCLK до установки dma_active[C] в 1 dma_sreq[C] и dma_waitonreq[C] установлены в 1 и dma_req[C] установлен в 0, контроллер выполняет 2 ^R передач с использованием первичной структуры управляющих данных. Затем без осуществления арбитража выполняет одну передачу, используя альтернативную структуру управляющих данных
17	Для типов циклов DMA, отличных от периферийного режима «Исполнение с изменением конфигурации», если за один такт HCLK до установки dma_active[C] в 1, а dma_sreq[C] и dma_req[C] установлены в 1, то приоритет предоставляется dma_req[c], и контроллер выполняет 2 ^R (или число передач, указанное в регистре n_minus_1) DMA передач. В периферийном режиме «Исполнение с изменением конфигурации», если за один такт HCLK до установки dma_active[C] в 1 dma_sreq[C] и dma_req[C] установлены в 1, то приоритет предоставляется dma_req[c], и контроллер выполняет 2 ^R передач с использованием первичной структуры управляющих данных, затем без осуществления арбитража выполняет передачи с номерами меньше, чем значение 2 ^R (или чем число передач, указанное в регистре n_minus_1), используя альтернативную структуру управляющих данных
18	Когда chnl_req_mask_set[C] установлен в 1, контроллер игнорирует запросы по dma_sreq[C] и dma_req[C]

*) Необходимо с осторожностью устанавливать эти разряды. Если значение, указанное в регистре n_minus_1 меньше, чем значение 2^R, то контроллер не очистит разряды chnl_useburst_set и поэтому запросы по dma_sreq[C] будут маскированы. Если периферия не устанавливает dma_req[C] в состояние 1, то контроллер никогда не выполнит необходимых передач.

При отключении канала контроллер осуществляет DMA передачи согласно правилам, представленным в таблице ниже.

Таблица 392 – Правила осуществления DMA передач при «запрещенных» каналах

Правило	Описание
19	Если dma_req[C] установлен в 1, то контроллер устанавливает dma_done[C] в 1. Это позволяет контроллеру показать центральному процессору запрос готовности, даже если канал выключен (запрещен)
20	Если dma_sreq[C] установлен в 1, то контроллер устанавливает dma_done[C] в 1 при условии dma_waitonreq[C] в 1 и chnl_useburst_set[C] в состоянии 0. Это позволяет контроллеру показать центральному процессору запрос готовности, даже если канал выключен (запрещен)
21	dma_active[C] всегда удерживается в состоянии 0

1.4.2 Диаграммы работы контроллера DMA

Данный раздел описывает следующие примеры функционирования контроллера с использованием правил обмена данными (Таблица 391):

- импульсный запрос на обработку;
- запрос по уровню на обработку;
- флаги завершения;
- флаги ожидания запроса на обработку.

Примечание – Все диаграммы, показанные на рисунках 83 – 86, подразумевают следующее:

Импульсный запрос на обработку

Рисунок 83 показывает временную диаграмму работы контроллера DMA при получении импульсного запроса от периферии.

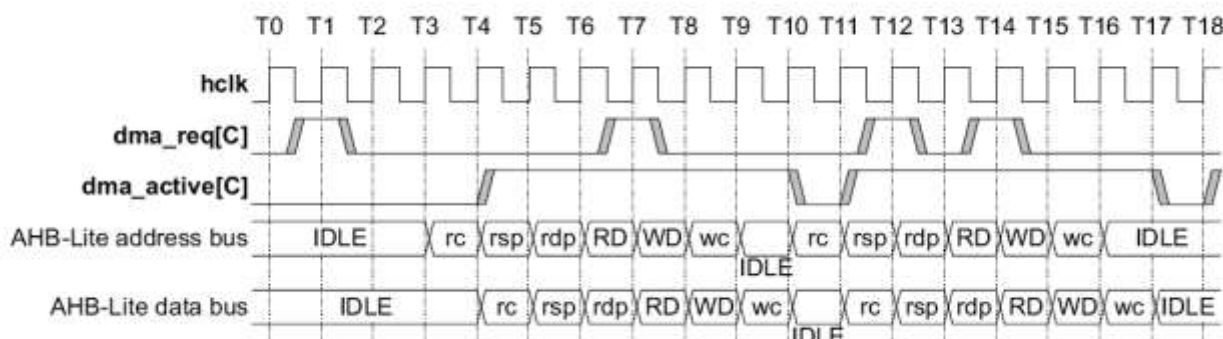


Рисунок 83 – Диаграмма работы при получении импульсного запроса от периферийного блока

Таблица 393 – Пояснения к диаграмме работы при получении импульсного запроса

T1	Контроллер обнаружил запрос на обработку по каналу C (см. правило 1) при условии, что <code>chnl_req_mask_set[C]</code> находится в состоянии 0 (см. правило 18)
T4	Контроллер устанавливает <code>dma_active[C]</code> (см. правила 2 и 3) и начинает DMA передачи по каналу C
T4-T7	Контроллер считывает управляющие данные канала, где: rc – чтение настроек канала, <code>channel_cfg</code> ; rsp – чтение указателя адреса окончания данных источника, <code>src_data_end_ptr</code> ; rdp – чтение указателя адреса окончания данных приемника, <code>dst_data_end_ptr</code>
T7	При установленном <code>dma_active[C]</code> в 1 и при условии, что <code>chnl_req_mask_set[C]</code> находится в состоянии 0, контроллер обнаруживает импульс запроса на обработки по каналу C (см. правило 7). Контроллер обработает этот запрос в течение следующего арбитража
T7-T9	Контроллер выполняет передачу DMA по каналу C, где: RD – чтение данных; WD – запись данных
T9-T10	Контроллер осуществляет запись настроек канала, <code>channel_cfg</code> , где wc – запись настроек канала, <code>channel_cfg</code>
T10	Контроллер сбрасывает сигнал <code>dma_active[C]</code> , что указывает на окончание передачи DMA (см. правило 4)
T10-T11	Контроллер удерживает <code>dma_active[C]</code> в 0 как минимум на один такт HCLK (см. правило 5)
T11	Если канал C имеет более высокий приоритет, то контроллер устанавливает <code>dma_active[C]</code> , так как ранее на такте T7 был получен запрос на обработку (см. правила 2 и 3)
T12	При установленном <code>dma_active[C]</code> в 1 и при условии, что <code>chnl_req_mask_set[C]</code> находится в состоянии 0, контроллер обнаруживает импульс запроса на обработку по каналу C (см. правило 7). Контроллер обработает этот запрос в течение следующего арбитража
T14	Контроллер игнорирует запрос по каналу C из-за отложенного запроса, полученного на такте T12
T17	Контроллер сбрасывает сигнал <code>dma_active[C]</code> , что указывает на окончание передачи DMA (см. правило 4)
T17-T18	Контроллер удерживает <code>dma_active[C]</code> как минимум на один такт HCLK (см. правило 5)
T18	Если канал C имеет более высокий приоритет, то контроллер устанавливает <code>dma_active[C]</code> , так как ранее на такте T12 был получен запрос на обработку (см. правила 2 и 3)

Запрос на обработку по уровню.

Рисунок 84 показывает временную диаграмму работы контроллера DMA при получении от периферии запроса на обработку по уровню.

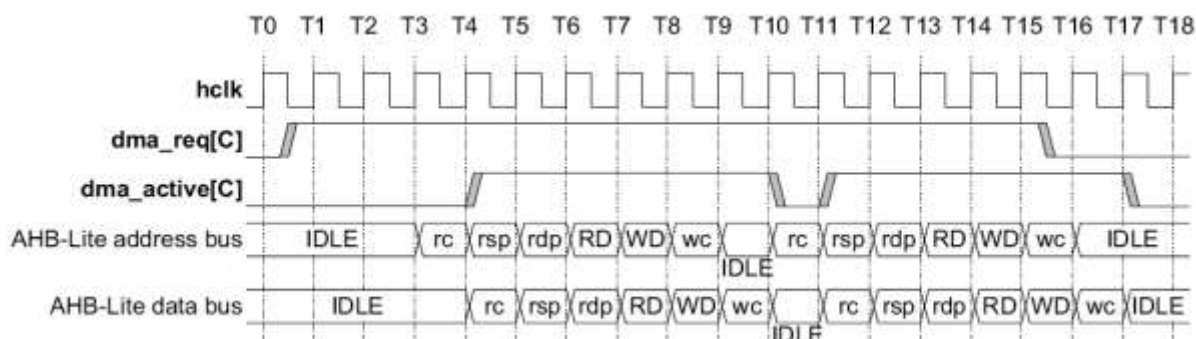


Рисунок 84 – Диаграмма работы при получении запроса на обработку по уровню

Таблица 394 – Пояснения к диаграмме работы при получении запроса на обработку по уровню

T1	Контроллер обнаружил запрос на обработку по каналу C (Таблица 391, правило 1) при условии, что <code>chnl_req_mask_set[C]</code> находится в состоянии 0 (см. правило 18)
T4	Контроллер устанавливает <code>dma_active[C]</code> (см. правила 2 и 3) и начинает DMA передачи по каналу C
T4-T7	Контроллер считывает управляющие данные канала, где: rc – чтение настроек канала, <code>channel_cfg</code> ; rsp – чтение указателя адреса окончания данных источника, <code>src_data_end_ptr</code> ; rdp – чтение указателя адреса окончания данных приемника, <code>dst_data_end_ptr</code> .
T7-T9	Контроллер выполняет передачу DMA по каналу C, где: RD – чтение данных; WD – запись данных.
T9-T10	Контроллер осуществляет запись настроек канала, <code>channel_cfg</code> , где wc – запись настроек канала, <code>channel_cfg</code>
T10	Контроллер сбрасывает сигнал <code>dma_active[C]</code> , что указывает на окончание передачи DMA (см. правило 4). Контроллер обнаружил запрос на обработку по каналу C (см. правило 1) при условии, что <code>chnl_req_mask_set[C]</code> находится в состоянии 0 (см. правило 18).
T10-T11	Контроллер удерживает <code>dma_active[C]</code> в 0 как минимум на один такт HCLK (см. правило 5)
T11	Если канал C имеет более высокий приоритет, то контроллер устанавливает <code>dma_active[C]</code> и начинает вторую DMA передачу по каналу C
T11-T14	Контроллер считывает управляющие данные канала
T14-T16	Контроллер выполняет передачу DMA по каналу C
T15-T16	Периферийный блок обнаруживает, что передача DMA началась и сбрасывает <code>dma_req[C]</code>
T16-T17	Контроллер осуществляет запись настроек канала <code>channel_cfg</code>
T17	Контроллер сбрасывает сигнал <code>dma_active[C]</code> , что указывает на окончание передачи DMA (см. правило 4)

При использовании запроса на обработку по уровню, периферийный блок может не обладать достаточным быстродействием, чтобы вовремя снять сигнал запроса, в этом случае он должен установить сигнал `dma_stall`. Установка сигнала `dma_stall` предотвращает повторение выполненной передачи.

Флаги завершения

Рисунок 85 демонстрирует функционирование сигнала (флага) dma_done[] при следующих условиях:

- dma_stall и dma_waitonreq[] находятся в состоянии 0;
- dma_stall установлен в 1;
- dma_waitonreq[] установлен в 1.

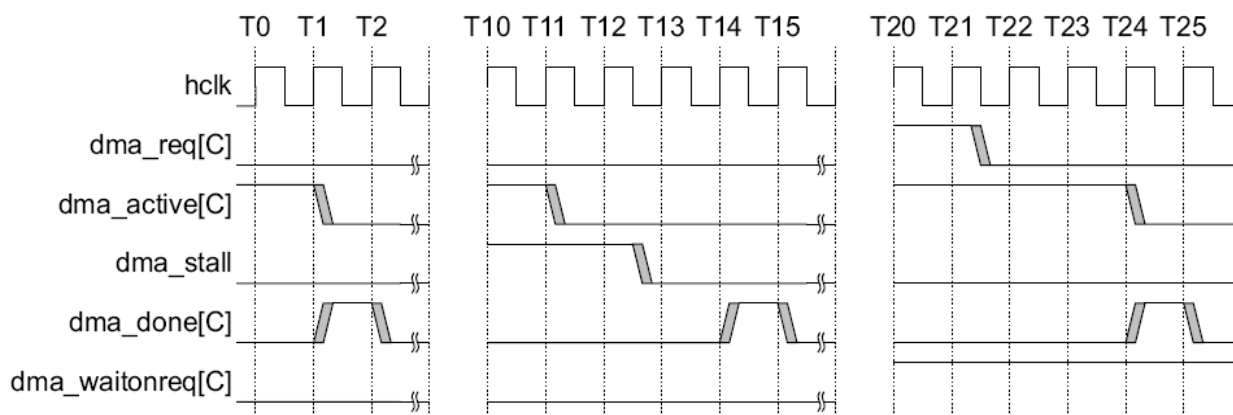


Рисунок 85 – Диаграммы функционирования dma_done

Таблица 395 – Пояснения функционирования dma_done, такты от T0 до T2

T1	Контроллер сбрасывает сигнал dma_active[C], что указывает на окончание передачи DMA (см. правило 4).
T1-T2	Контроллер завершает цикл DMA, и если cycle_ctrl[2] установлен в 0, то устанавливает в 1 dma_done[C] на один такт HCLK (см. правила 8 и 9). Для других разрешенных каналов сигнал dma_done[C] останется в состоянии 0 (см. правило 6).

Таблица 396 – Пояснения функционирования dma_done, такты от T10 до T15

T11	Контроллер сбрасывает сигнал dma_active[C], что указывает на окончание передачи DMA (см. правило 4).
T12-T13	Периферийный блок сбрасывает сигнал dma_stall.
T14-T15	Контроллер завершает цикл DMA, и если cycle_ctrl[2] установлен в 0, то устанавливает в 1 dma_done[C] на один такт HCLK (см. правила 8 и 9). Для других разрешенных каналов сигнал dma_done[C] останется в состоянии 0 (см. правило 6).

Примечание к T11 – Контроллер не устанавливает сигнал dma_done[C], так как сигнал dma_stall установлен в 1 в предшествующем такте HCLK (см. правила 9 и 12).

Таблица 397 – Пояснения функционирования dma_done, такты от T20 до T25

T20	Контроллер выполнил передачу DMA, но из-за установленного в 1 dma_waitonreq[C] он должен ожидать сброса в 0 сигнала dma_req[C], перед тем как сбросить dma_active[C] (см. правило 11) и установить dma_done[C] (см. правило 9).
T21-T25	Периферийный блок сбрасывает dma_req[C].
T24	Контроллер сбрасывает сигнал dma_active[C], что указывает на окончание передачи DMA (см. правило 4).
T24-T25	Контроллер завершает цикл DMA и, если cycle_ctrl[2] установлен в 0, то устанавливает в 1 dma_done[C] на один такт HCLK (см. правила 8 и 9). Для других разрешенных каналов сигнал dma_done[C] останется в состоянии 0 (см. правило 6)

Флаги ожидания запроса на обработку

Ниже приведены рисунки, которые демонстрируют примеры использования флагов ожидания запроса на обработку при выполнении 2^R передач и одиночных передач:

- диаграмма работы контроллера DMA при использовании периферией dma_waitonreq;
- диаграмма работы контроллера DMA при использовании периферией dma_waitonreq совместно с dma_sreq.

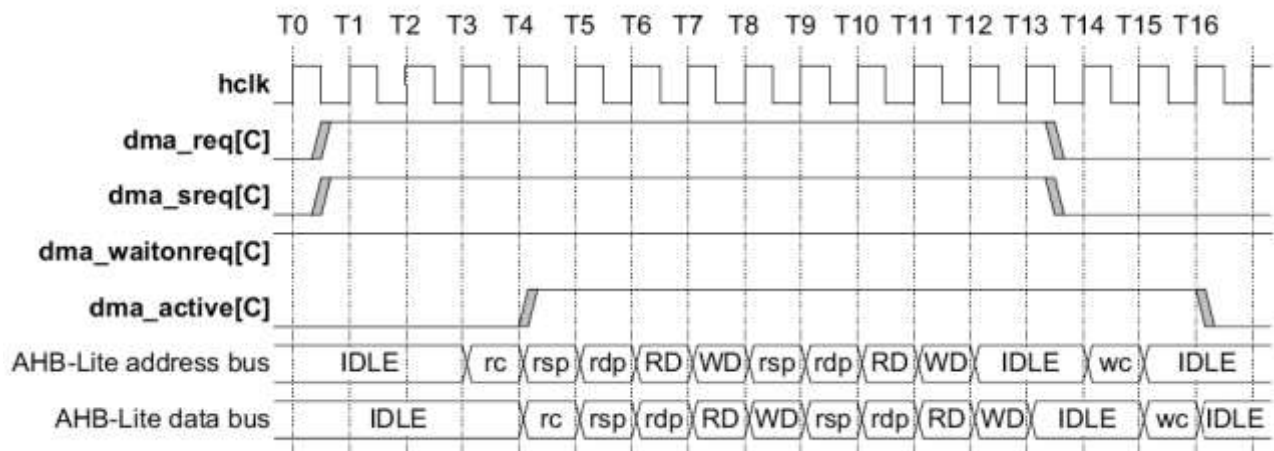


Рисунок 86 – Диаграмма работы контроллера DMA при использовании dma_waitonreq

**Таблица 398 – Пояснения работы контроллера DMA
при использовании dma_waitonreq**

T0-T16	Периферийный блок должен оставлять состояние dma_waitonreq[C] постоянно (см. правило 10).
T0-T1	Контроллер обнаружил запрос на обработку по каналу C (см. правило 1) при условии, что chnl_req_mask_set[C] находится в состоянии 0 (см. правило 18).
T3-T4	Периферийный блок удерживает dma_req[C] и dma_sreq[C] в 1. Контроллер игнорирует dma_sreq[C] запрос и отвечает на dma_req[C] запрос (см. правила 16 и 17).
T4	Контроллер устанавливает dma_active[C] (см. правила 2 и 3) и начинает DMA передачи по каналу C
T4-T7	Контроллер считывает управляющие данные канала, где: rc – чтение настроек канала, channel_cfg; rsp – чтение указателя адреса окончания данных источника, src_data_end_ptr; rdp – чтение указателя адреса окончания данных приемника, dst_data_end_ptr.
T7-T9	Контроллер выполняет передачу DMA по каналу C, где: RD – чтение данных; WD – запись данных.
T9-T11	Контроллер считывает 2 указателя адреса окончания данных rsp и rdp.
T11-T13	Периферийный блок сбрасывает сигналы dma_req[C] и dma_sreq[C].
T15-T16	Контроллер осуществляет запись настроек канала, channel_cfg, где wc – запись настроек канала, channel_cfg.
T16	Контроллер сбрасывает сигнал dma_active[C], что указывает на окончание передачи DMA (см. правило 11). Контроллер устанавливает значение по чтению регистра chnl_useburst_set[C] в 0, если количество оставшихся передач менее 2 ^R (см. правило 15).

Рисунок 87 показывает работу контроллера DMA при установке dma_waitonreq в 1 и выполнении одиночной DMA передачи.

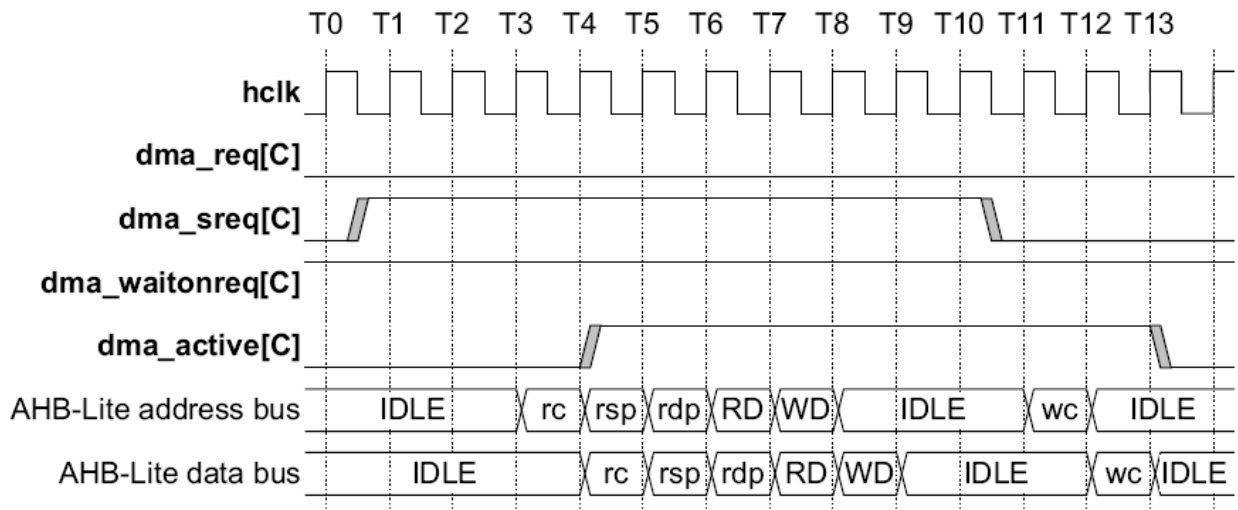


Рисунок 87 – Работа DMA при использовании dma_waitonreq совместно с dma_sreq

Таблица 399 – Пояснения работы DMA при использовании dma_waitonreq совместно с dma_sreq

T0-T13	Периферийный блок должен оставлять состояние dma_waitonreq[C] постоянно (см. правило 10).
T0-T1	Контроллер обнаружил запрос на обработку по каналу C (см. правило 1) при условии, что chnl_useburst_set[C] находится в состоянии 0 (см. правила 13 и 14).
T3-T4	Контроллер отвечает на dma_sreq[C] запрос (см. правила 16).
T4	Контроллер устанавливает dma_active[C] (см. правила 2 и 3) и начинает DMA передачи по каналу C.
T4-T7	Контроллер считывает управляющие данные канала, где: rc – чтение настроек канала, channel_cfg; rsp – чтение указателя адреса окончания данных источника, src_data_end_ptr; rdp – чтение указателя адреса окончания данных приемника, dst_data_end_ptr.
T7-T9	Контроллер выполняет передачу DMA по каналу C, где: RD – чтение данных; WD – запись данных. Это запрос в ответ на dma_sreq[], таким образом, R=0 и, следовательно, контроллер исполнит 1 DMA передачу.
T10-T11	Периферийный блок сбрасывает сигнал dma_sreq[C].
T12-T13	Контроллер осуществляет запись настроек канала, channel_cfg, где wc – запись настроек канала, channel_cfg.
T13	Контроллер сбрасывает сигнал dma_active[C], что указывает на окончание передачи DMA (см. правило 11).

1.4.3 Правила арбитража DMA

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

Контроллер имеет 4 разряда, которые определяют количество транзакций по шине АНВ до повторения арбитража. Эти разряды задают степень R числа 2; изменение R напрямую устанавливает периодичность арбитража как 2 в степени R. Для примера, если R равно 4, то арбитраж будет проводиться через каждые 16 передач DMA.

Таблица 400 показывает возможную периодичность арбитража.

Таблица 400 – Периодичность арбитража в единицах передач по шине АНВ

Значение R	Периодичность арбитража каждые x передач DMA
4'b0000	1
4'b0001	2
4'b0010	4
4'b0011	8
4'b0100	16
4'b0101	32
4'b0110	64
4'b0111	128
4'b1000	256
4'b1001	512
4'b1010 – 4'b1111	1024

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

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

Разряды степени R числа 2 находятся в структуре управляющих данных канала. Местонахождение этих разрядов описано в разделе «Управляющие данные канала».

1.4.4 Приоритет

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

- номер канала;
- уровень приоритета, присвоенного каналу.

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

Канал номер 0 имеет высший уровень приоритета, уровень приоритета снижается с увеличением номера канала. Таблица 401 показывает уровень приоритета каналов DMA в порядке его уменьшения.

Таблица 401 – Уровень приоритета каналов DMA

Номер канала	Установка уровня приоритета	Уровень приоритета в порядке его уменьшения
0	Высокий	Наивысший уровень приоритета
1	Высокий	-
2	Высокий	-
-	Высокий	-
-	Высокий	-
-	Высокий	-
30	Высокий	-
31	Высокий	-
0	По умолчанию (низкий)	-
1	По умолчанию (низкий)	-
2	По умолчанию (низкий)	-
-	По умолчанию (низкий)	-
-	По умолчанию (низкий)	-
-	По умолчанию (низкий)	-
30	По умолчанию (низкий)	-
31	По умолчанию (низкий)	Низший уровень приоритета

После окончания цикла DMA контроллер выбирает следующий для обслуживания канал из всех включенных каналов DMA.

Рисунок 88 показывает процесс выбора следующего канала для обслуживания.

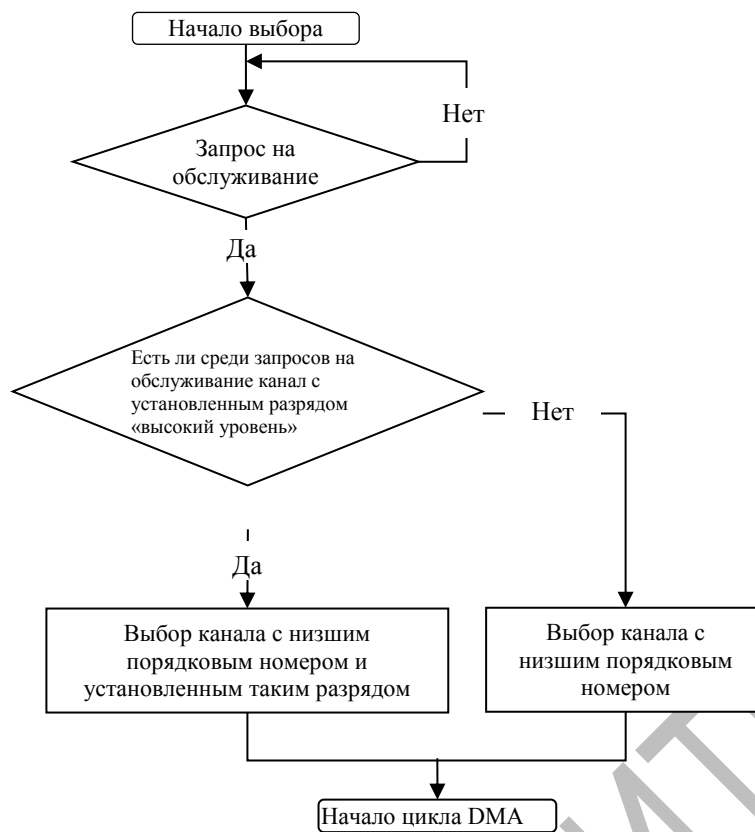


Рисунок 88 – Алгоритм выбора следующего канала для обслуживания

Начало выбора.
Есть ли запрос на обслуживание.
Есть ли среди запросов на обслуживание канал с установленным разрядом «высокий уровень».
Выбор канала с низшим порядковым номером и установленным таким разрядом.
Выбор канала с низшим порядковым номером.
Начало цикла DMA.

1.4.5 Типы циклов DMA

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

Таблица 402 – Типы циклов DMA

cycle_ctrl	Описание
3'b000	Структура управляющих данных канала в запрещенном состоянии
3'b001	Обычный цикл DMA
3'b010	Авто-запрос
3'b011	Режим пинг-понг
3'b100	Работа с памятью в режиме «Исполнение с изменением конфигурации» с использованием первичных управляющих данных канала
3'b101	Работа с памятью в режиме «Исполнение с изменением конфигурации» с использованием альтернативных управляющих данных канала
3'b110	Работа с периферией в режиме «Исполнение с изменением конфигурации» с использованием первичных управляющих данных канала
3'b111	Работа с периферией в режиме «Исполнение с изменением конфигурации» с использованием альтернативных управляющих данных канала

Примечание – Разряды `cycle_ctrl` находятся в области памяти, отведенной под `channel_cfg` – см. раздел «Настройка управляющих данных канала».

Для всех типов циклов DMA повторный арбитраж происходит после 2^R передач DMA. Если установить длинный период арбитража на низкоприоритетном канале, то это заблокирует все запросы на обработку от других каналов до тех пор, пока не будут выполнены 2^R передач DMA по данному каналу. Поэтому, устанавливая значение `R`, необходимо учитывать, что это может привести к повышенному времени отклика на запрос на обработку от высокоприоритетных каналов.

Данный раздел описывает следующие типы циклов DMA:

- недействительный;
- основной;
- авто-запрос;
- «пинг-понг»;
- работа с памятью в режиме «исполнение с изменением конфигурации»;
- работа с периферией в режиме «исполнение с изменением конфигурации».

Недействительный

После окончания цикла DMA контроллер устанавливает тип цикла в значение «недействительный» для предотвращения повтора выполненного цикла DMA.

Основной

В этом режиме контроллер работает только с основными или альтернативными управляющими данными канала. После того, как разрешена работа канала, и контроллер получил запрос на обработку, цикл DMA выглядит следующим образом:

1. Контроллер выполняет 2^R передач. Если число оставшихся передач 0, контроллер переходит к шагу 3.
2. Осуществление арбитража:

- если высокоприоритетный канал выдает запрос на обработку, то контроллер начинает обслуживание этого канала;

- если периферийный блок или программное обеспечение выдает запрос на обработку (повторный запрос на обработку по каналу), то контроллер переходит к шагу 1.

3. Контроллер устанавливает `dma_done[C]` в состояние 1 на один такт сигнала HCLK. Это указывает центральному процессору на завершение цикла DMA.

Авто-запрос

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

Контроллер позволяет выбрать для использования первичную или альтернативную структуру управляющих данных канала. После того как разрешена работа канала и контроллер получил запрос на обработку, цикл DMA выглядит следующим образом:

1. Контроллер выполняет 2^R передач для канала C. Если число оставшихся передач 0, контроллер переходит к шагу 3.

2. Осуществление арбитража:

- если высокоприоритетный канал выдает запрос на обработку, то контроллер начинает обслуживание этого канала;

- если периферийный блок или программное обеспечение выдает запрос на обработку (повторный запрос на обработку по каналу), то контроллер переходит к шагу 1.

3. Контроллер устанавливает `dma_done[C]` в состояние 1 на один такт сигнала HCLK. Это указывает центральному процессору на завершение цикла DMA.

Пинг-понг

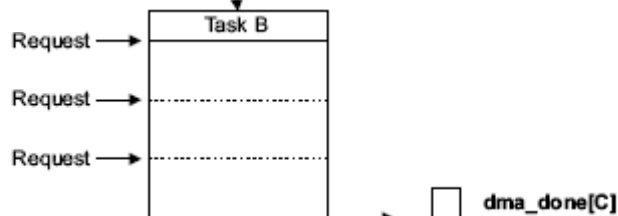
В данном режиме контроллер выполняет цикл DMA, используя одну из структур управляющих данных, а затем выполняет еще один цикл DMA, используя другую структуру управляющих данных. Контроллер выполняет циклы DMA с переключением структур до тех пор, пока не считает «недействительную» структуру данных или пока процессор не запретит работу канала.

Рисунок 89 демонстрирует пример функционирования контроллера в режиме «пинг-понг».

Шаг А. Первичная структура,
cycle_ctrl=b011, $2^R = 4$, N=6



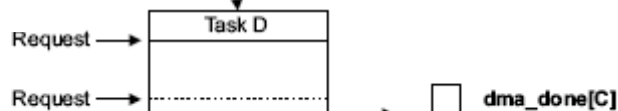
Шаг В. Альтернативная структура,
cycle_ctrl=b011, $2^R = 4$, N=12



Шаг С. Первичная структура,
cycle_ctrl=b011, $2^R = 2$, N=2



Шаг D. Альтернативная структура,
cycle_ctrl=b011, $2^R = 4$, N=5



Шаг Е. Первичная структура,
cycle_ctrl=b011, $2^R = 4$, N=7



Конец. Альтернативная структура,
cycle_ctrl=b000

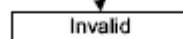


Рисунок 89 – Пример функционирования контроллера в режиме «пинг-понг»

Пояснения к рисунку

Шаг А

Процессор устанавливает первичную структуру управляющих данных для шага А.

Процессор устанавливает альтернативную структуру управляющих данных для шага В. Это позволит контроллеру переключиться к шагу В незамедлительно после выполнения шага А, при условии, что контроллер не получит запрос на обработку от высокоприоритетного канала.

Контроллер получает запрос и выполняет 4 передачи DMA.

Контроллер выполняет арбитраж. После получения запроса на обработку от этого же канала, контроллер продолжает цикл в ситуации отсутствия высокоприоритетных запросов.

Контроллер выполняет оставшиеся 2 передачи DMA.

Контроллер устанавливает `dma_done[C]` в состояние 1 на один такт сигнала синхронизации HCLK и входит в процедуру арбитража

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

После получения нового запроса на обработку от канала при условии его наивысшего приоритета выполняется шаг В:

Шаг В Контроллер выполняет 4 передачи DMA.
Контроллер выполняет арбитраж. После получения запроса на обработку от этого же канала контроллер продолжает цикл в ситуации отсутствия высокоприоритетных запросов.
Контроллер выполняет 4 передачи DMA.
Контроллер выполняет арбитраж. После получения запроса на обработку от этого же канала контроллер продолжает цикл в ситуации отсутствия высокоприоритетных запросов.
Контроллер выполняет оставшиеся 4 передачи DMA.
Контроллер устанавливает `dma_done[C]` в состояние 1 на один такт сигнала синхронизации HCLK и входит в процедуру арбитража.

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

После получения нового запроса на обработку от канала при условии его наивысшего приоритета выполняется шаг С:

Шаг С Контроллер выполняет 2 передачи DMA.
Контроллер устанавливает `dma_done[C]` в состояние 1 на один такт сигнала синхронизации HCLK и входит в процедуру арбитража.

После выполнения шага С процессор может установить первичные управляющие данные канала для шага E.

После получения нового запроса на обработку от канала при условии его наивысшего приоритета выполняется шаг D:

Шаг D Контроллер выполняет 4 передачи DMA.
Контроллер выполняет арбитраж. После получения запроса на обработку от этого же канала контроллер продолжает цикл в ситуации отсутствия высокоприоритетных запросов.
Контроллер выполняет оставшуюся передачу DMA.
Контроллер устанавливает `dma_done[C]` в состояние 1 на один такт сигнала синхронизации HCLK и входит в процедуру арбитража.

После получения нового запроса на обработку от канала при условии его наивысшего приоритета выполняется шаг E:

Шаг E Контроллер выполняет 4 передачи DMA.
Контроллер выполняет арбитраж. После получения запроса на обработку от этого же канала контроллер продолжает цикл в ситуации отсутствия высокоприоритетных запросов.
Контроллер выполняет оставшиеся 3 передачи DMA.
Контроллер устанавливает `dma_done[C]` в состояние 1 на один такт сигнала синхронизации HCLK и входит в процедуру арбитража.

Если контроллер получит новый запрос на обработку от данного канала и этот запрос будет самым приоритетным, контроллер предпримет попытку выполнения следующего шага. Однако из-за того, что процессор не установил альтернативные управляющие данные, и по окончании шага D контроллер установил `cycle_ctrl` в состояние `b000`, передачи DMA прекращаются.

Примечание – Для прерывания цикла DMA, исполняемого в режиме «пинг-понг», также возможен перевод режима работы контроллера на шаге E в режим «Основной цикл DMA» путем установки `cycle_ctrl` в `3'b001`.

Режим работы с памятью «исполнение с изменением конфигурации»

В данном режиме контроллер, получая начальный запрос на обработку, выполняет 4 передачи DMA, используя первичные управляющие данные. По окончании этих передач контроллер начинает цикл DMA, используя альтернативные управляющие данные. Затем контроллер выполняет еще 4 передачи DMA, используя первичные управляющие данные. Контроллер продолжает выполнять циклы DMA, меняя структуры управляющих данных, пока не произойдет одно из следующих условий:

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

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

Контроллер устанавливает флаг `dma_done[C]` в этом режиме работы только тогда, когда передача DMA заканчивается с использованием основного цикла.

В данном режиме контроллер использует первичные управляющие данные для программирования альтернативных управляющих данных. Таблица 403 перечисляет области памяти `channel_cfg`, те которые должны быть определены константами, и те, значения которых определяются пользователем.

Таблица 403 – Channel_cfg для первичной структуры управляющих данных в режиме работы с памятью «исполнение с изменением конфигурации»

Разряды	Обозначение	Значение	Описание
Области с константными значениями			
31, 30	<code>dst_inc</code>	<code>2'b10</code>	Контроллер производит инкремент адреса пословно
29, 28	<code>dst_size</code>	<code>2'b10</code>	Контроллер осуществляет передачу пословно
27, 26	<code>src_inc</code>	<code>2'b10</code>	Контроллер производит инкремент адреса пословно
25, 24	<code>src_size</code>	<code>2'b10</code>	Контроллер осуществляет передачу пословно
17...14	<code>R_power</code>	<code>4'b0010</code>	Контроллер выполняет 4 передачи DMA

3	next_useburst	1'b0	Для данного режима этот разряд должен быть равен 0
2...0	cycle_ctrl	3'b100	Контролер работает в режиме работы с периферией «исполнение с изменением конфигурации»
Области со значениями, определяемыми пользователем			
23...21	dst_prot_ctrl	-	Определяет состояние HPROT при записи данных в приемник
20...18	src_prot_ctrl	-	Определяет состояние HPROT при чтении данных из источника
13...4	n_minus_1	N*	Настраивает контроллер на выполнение N передач DMA, где N кратно 4

* – Так как разряды R_power установлены в состояние 2, необходимо задавать значение N кратное 4. Число равное N/4 это количество раз, которое нужно настраивать альтернативные управляющие данные.

Рисунок 90 демонстрирует пример функционирования в режиме работы с памятью «исполнение с изменением конфигурации».

Инициализация:

1. Настройка первичных управляющих данных для разрешения копирования A, B, C и D: cycle_ctrl=3'b100, $2^R=4$, $N=16$.
2. Запись первичных данных в память с использованием структуры, показанной в таблице ниже.

	src_data_end_ptr	dst_data_end_ptr	channel_cfg	Unused
Data for Task A	0x0A000000	0x0AE00000	cycle_ctrl = b101, $2^R = 4$, N = 3	0XXXXXXXXX
Data for Task B	0x0B000000	0x0BE00000	cycle_ctrl = b101, $2^R = 2$, N = 8	0XXXXXXXXX
Data for Task C	0x0C000000	0x0CE00000	cycle_ctrl = b101, $2^R = 8$, N = 5	0XXXXXXXXX
Data for Task D	0x0D000000	0x0DE00000	cycle_ctrl = b001, $2^R = 4$, N = 4	0XXXXXXXXX

Memory scatter-gather transaction:

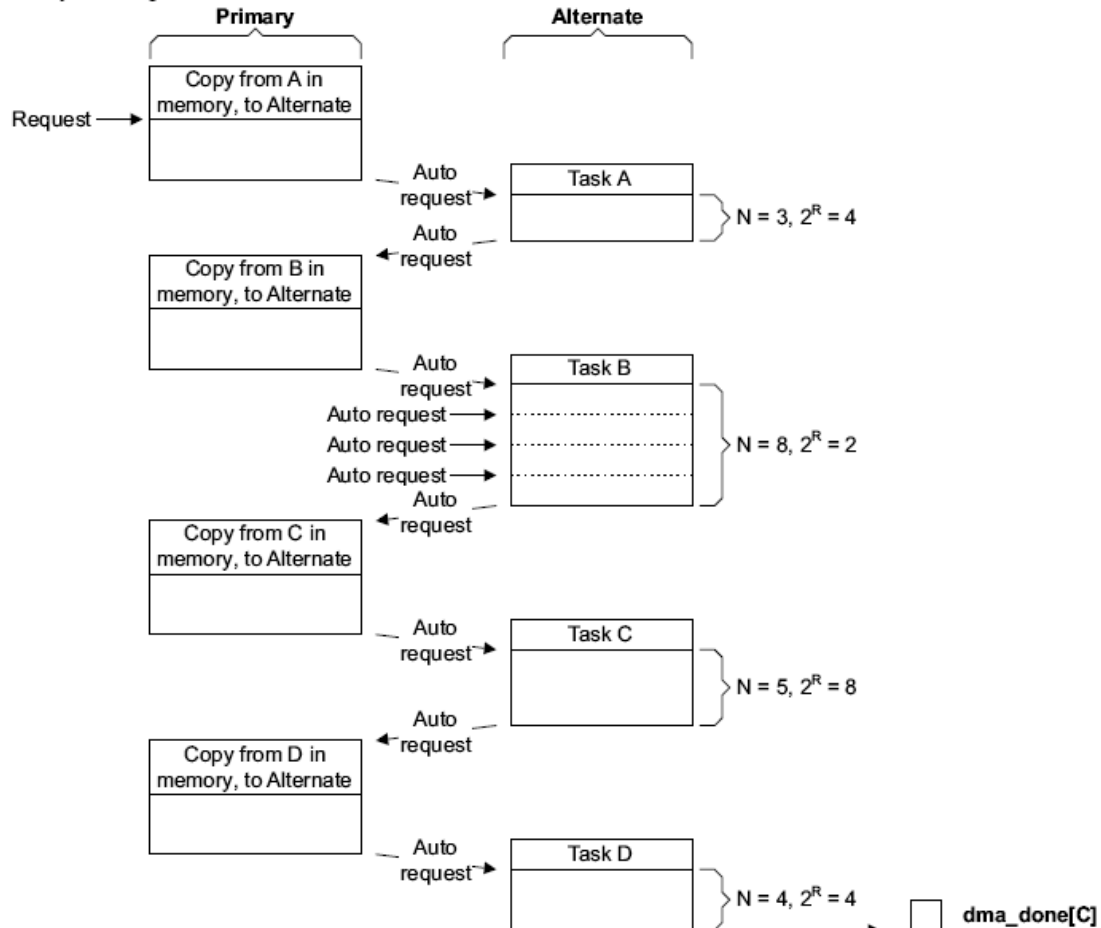


Рисунок 90 – Пример работы DMA в режиме с «Исполнением с изменением конфигурации»

Инициализация:

1. Процессор настраивает первичную структуру управляющих данных для работы в режиме работы с памятью «исполнение с изменением конфигурации» путем установки cycle_ctrl в 3b'100. Так как управляющие данные канала состоят из 4 слов, необходимо установить 2^R в 4. В этом примере количество передач равно 4 и поэтому N установлен в 16.
2. Процессор записывает управляющие данные для шагов A, B, C, D в область памяти с адресом, указанным в src_data_end_ptr.
3. Процессор разрешает работу канала DMA.

Передачи в данном режиме начинают исполняться при получении контроллером запроса на обслуживание по dma_req[] или запроса от процессора. Порядок выполнения следующий:

Первичная, копирование A

По получению запроса на обслуживание контроллер выполняет 4 передачи DMA. Эти передачи записывают альтернативную структуру управляющих данных для шага А.

Контроллер генерирует автозапрос для канала, после чего проводит процедуру арбитража.

Шаг А

Контроллер выполняет шаг А. По окончании контроллер генерирует автозапрос для канала и проводит процедуру арбитража.

Первичная, копирование В

Контроллер выполняет 4 передачи DMA. Эти передачи записывают альтернативную структуру управляющих данных для шага В.

Контроллер генерирует автозапрос для канала, после чего проводит процедуру арбитража.

Шаг В

Контроллер выполняет шаг В. По окончании контроллер генерирует автозапрос для канала и проводит процедуру арбитража.

Первичная, копирование С

Контроллер выполняет 4 передачи DMA. Эти передачи записывают альтернативную структуру управляющих данных для шага С.

Контроллер генерирует автозапрос для канала, после чего проводит процедуру арбитража.

Шаг С

Контроллер выполняет шаг С. По окончании контроллер генерирует автозапрос для канала и проводит процедуру арбитража.

Первичная, копирование D

Контроллер выполняет 4 передачи DMA. Эти передачи записывают альтернативную структуру управляющих данных для шага D.

Контроллер устанавливает `cycle_ctrl` первичных управляющих данных в `3'b000` для индикации о том, что эта структура управляющих данных является «неправильной».

Контроллер генерирует автозапрос для канала, после чего проводит процедуру арбитража.

Шаг D

Контроллер выполняет шаг D, используя основной цикл DMA.

Контроллер устанавливает флаг `dma_done[C]` в состояние 1 на один такт сигнала HCLK и входит в процедуру арбитража.

Режим работы с периферией «исполнение с изменением конфигурации»

В данном режиме контроллер, получая начальный запрос на обработку, выполняет 4 передачи DMA, используя первичные управляющие данные. По окончании этих передач контроллер начинает цикл DMA, используя альтернативные управляющие данные без осуществления арбитража и не устанавливая сигнал `dma_active[C]` в 0.

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

После того, как этот цикл завершился, контроллер выполняет арбитраж и по получении запроса на обслуживание от периферии, имеющего наивысший приоритет, он

выполняет еще 4 передачи DMA, используя первичные управляющие данные. По окончании этих передач контроллер начинает цикл DMA, используя альтернативные управляющие данные без осуществления арбитража и не устанавливая сигнал dma_active[C] в 0.

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

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

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

Контроллер устанавливает флаг dma_done[C] в этом режиме работы только тогда, когда передача DMA заканчивается с использованием основного цикла.

В данном режиме контроллер использует первичные управляющие данные для программирования альтернативных управляющих данных. Таблица 404 перечисляет области памяти channel_cfg, которые должны быть определены константами, и те области, значения которых определяются пользователем.

Таблица 404 – Channel_cfg для первичной структуры управляющих данных в режиме работы с периферией «Исполнение с изменением конфигурации»

Разряды	Обозначение	Значение	Описание
Области с константными значениями			
31, 30	dst_inc	2'b10	Контроллер производит инкремент адреса пословно
29, 28	dst_size	2'b10	Контроллер осуществляет передачу пословно
27, 26	src_inc	2'b10	Контроллер производит инкремент адреса пословно
25, 24	src_size	2'b10	Контроллер осуществляет передачу пословно
17...14	R_power	4'b0010	Контроллер выполняет 4 передачи DMA
2...0	cycle_ctrl	3'b110	Контроллер работает в режиме работы с периферией «исполнение с изменением конфигурации»
Области со значениями, определяемыми пользователем			
23...21	dst_prot_ctrl	-	Определяет состояние HPROT при записи данных в приемник
20...18	src_prot_ctrl	-	Определяет состояние HPROT при чтении данных из источника
13...4	n_minus_1	N*	Настраивает контроллер на выполнение N передач DMA, где N кратно 4
3	next_useburst	-	При установке в 1, контроллер установит chnl_useburst_set[C] в 1 после выполнения передачи с альтернативной структурой

* Так как разряды R_power установлены в состояние 2, необходимо задавать значение N кратное 4. Число равное N/4 это количество раз, которое нужно настраивать альтернативные управляющие данные.

Рисунок 91 демонстрирует пример функционирования в режиме работы с периферией «исполнение с изменением конфигурации».

Инициализация:

1. Настройка первичных управляющих данных для разрешения копирования A, B, C и D:
cycle_ctrl=3'b110, $2^R=4$, N=16.
2. Запись первичных данных в память с использованием структуры, показанной в таблице ниже.

ПРЕДВАРИТЕЛЬНО

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

Peripheral scatter-gather transaction:

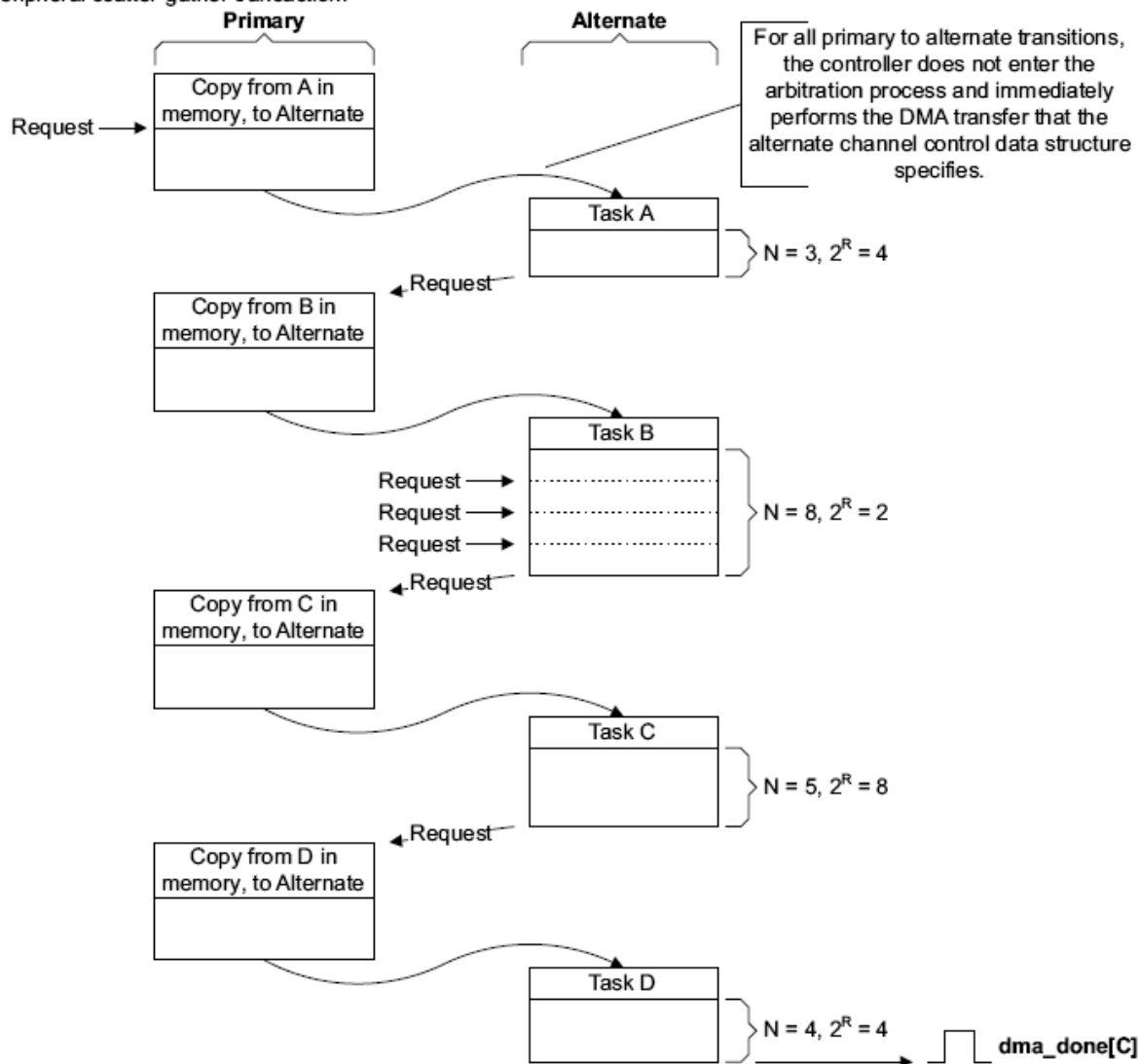


Рисунок 91 – Пример работы DMA в режиме с «Исполнением с изменением конфигурации»

Пояснения к рисунку

Инициализация:

1. Процессор настраивает первичную структуру управляющих данных для работы в режиме работы с периферией «исполнение с изменением конфигурации» путем установки cycle_ctrl в 3'b110. Так как управляющие данные канала состоят из 4 слов, необходимо установить 2^R в 4. В этом примере количество задач равно 4 и поэтому N установлено в 16.

2. Процессор записывает управляющие данные для шагов A, B, C, D в область памяти с адресом, указанным в src_data_end_ptr.

3. Процессор разрешает работу канала DMA.

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

Первичная, копирование из области А памяти

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

Шаг А

Контроллер выполняет шаг А.

По окончании контроллер проводит процедуру арбитража.

Первичная, копирование из области В памяти.

Контроллер выполняет 4 передачи DMA. Эти передачи записывают альтернативную структуру управляющих данных для шага В.

Шаг В

Контроллер выполняет шаг В. Для завершения задачи периферия должна установить последовательно 3 запроса.

По окончании контроллер проводит процедуру арбитража.

Первичная, копирование из области С памяти.

Контроллер выполняет 4 передачи DMA. Эти передачи записывают альтернативную структуру управляющих данных для шага С.

Шаг С

Контроллер выполняет шаг С.

По окончании контроллер проводит процедуру арбитража.

После выставления периферией нового запроса на обслуживание, при условии, что этот запрос является наиболее приоритетным, процесс продолжается следующим образом:

Первичная, копирование из области D памяти.

Контроллер выполняет 4 передачи DMA. Эти передачи записывают альтернативную структуру управляющих данных для шага D.

Контроллер устанавливает `cycle_ctrl` первичных управляющих данных в `3'b000` для индикации о том, что эта структура управляющих данных является «неправильной».

Шаг D

Контроллер выполняет шаг D, используя основной цикл DMA.

Контроллер устанавливает флаг `dma_done[C]` в состояние 1 на один такт сигнала HCLK и входит в процедуру арбитража.

Индикация ошибок

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

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

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

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

- процессор должен сравнить список выключенных каналов, полученный в шаге 1, с данными о каналах, которые недавно устанавливали флаги `dma_done[]`. Канал, по которому отсутствуют данные об установке флага `dma_done[]`, это и есть канал, с которым связана ошибка.

ПРЕДВАРИТЕЛЬНО

1.5 Структура управляющих данных канала

В системной памяти должна быть отведена область для хранения управляющих данных каналов. Системная память должна:

- предоставлять смежную область системной памяти, к которой контроллер и процессор имеют доступ;
- иметь базовый адрес, который целочисленно кратен общему размеру структуры управляющих данных канала.

Рисунок 92 показывает область памяти необходимую контроллеру для структур управляющих данных канала, при использовании всех 32 каналов и опциональной альтернативной структуры управляющих данных.

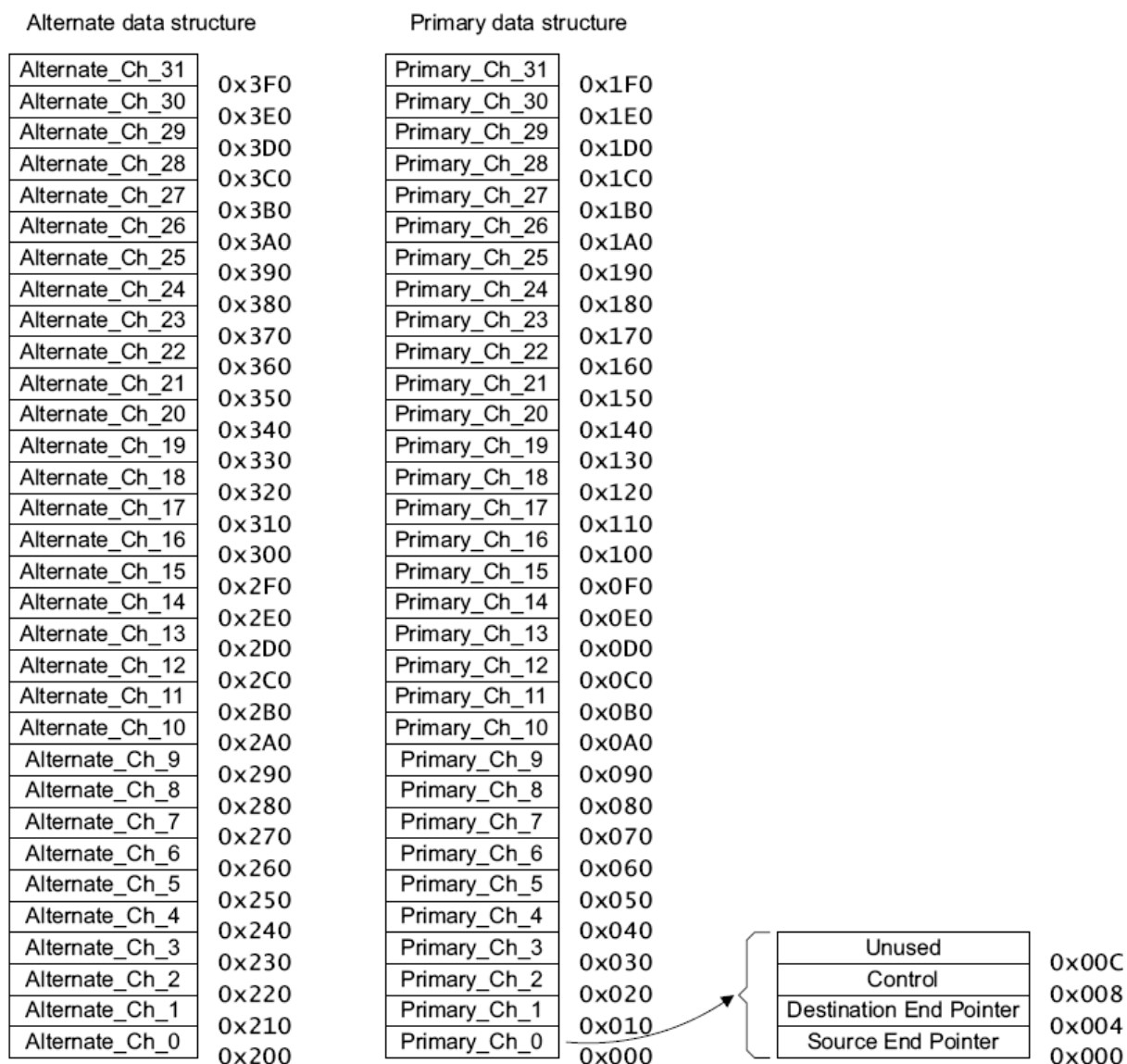


Рисунок 92 – Карта памяти для 32-х каналов, включая альтернативную структуру управляющих данных

Пример использует 1 Кбайт системной памяти. В этом примере контроллер использует младшие 10 разрядов адреса для доступа ко всем элементам структуры управляющих данных, и поэтому базовый адрес структуры должен быть 0xXXXXXX000, далее 0xXXXXXX400, далее 0xXXXXXX800, далее 0xXXXXXXC00.

Базовый адрес для первичной структуры управляющих данных устанавливается путем записи соответствующего значения в регистр ctrl_base_ptr.

Необходимый размер области системной памяти зависит от:

- количества каналов, используемых в контроллере;
- от того, используется или нет альтернативная структура управляющих данных.

Таблица 405 перечисляет разряды адреса, которые используются контроллером при доступе к различным элементам структуры управляющих данных, в зависимости от количества каналов, используемых в контроллере.

Таблица 405 – Разряды адреса, соответствующие элементам структуры управляющих данных

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

где

A выбирает одну из структур управляющих данных канала:

A = 0 выбирает первичную структуру управляющих данных;

A = 1 выбирает альтернативную структуру управляющих данных.

C[x:0] выбирает канал DMA.

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

0x0 выбирает указатель конца данных источника;

0x4 выбирает указатель конца данных приемника;

0x8 выбирает конфигурацию управляющих данных;

0xC контроллер не имеет доступа к этому адресу.

Если это необходимо, то возможно разрешить процессору использовать эти адреса в качестве системной памяти.

Примечание – Совсем не обязательно вычислять базовый адрес альтернативной структуры управляющих данных, так как регистр alt_ctrl_base_ptr содержит эту информацию.

Рисунок 93 демонстрирует пример реализации контроллера с использованием 3 каналов DMA и альтернативной структурой управляющих данных.

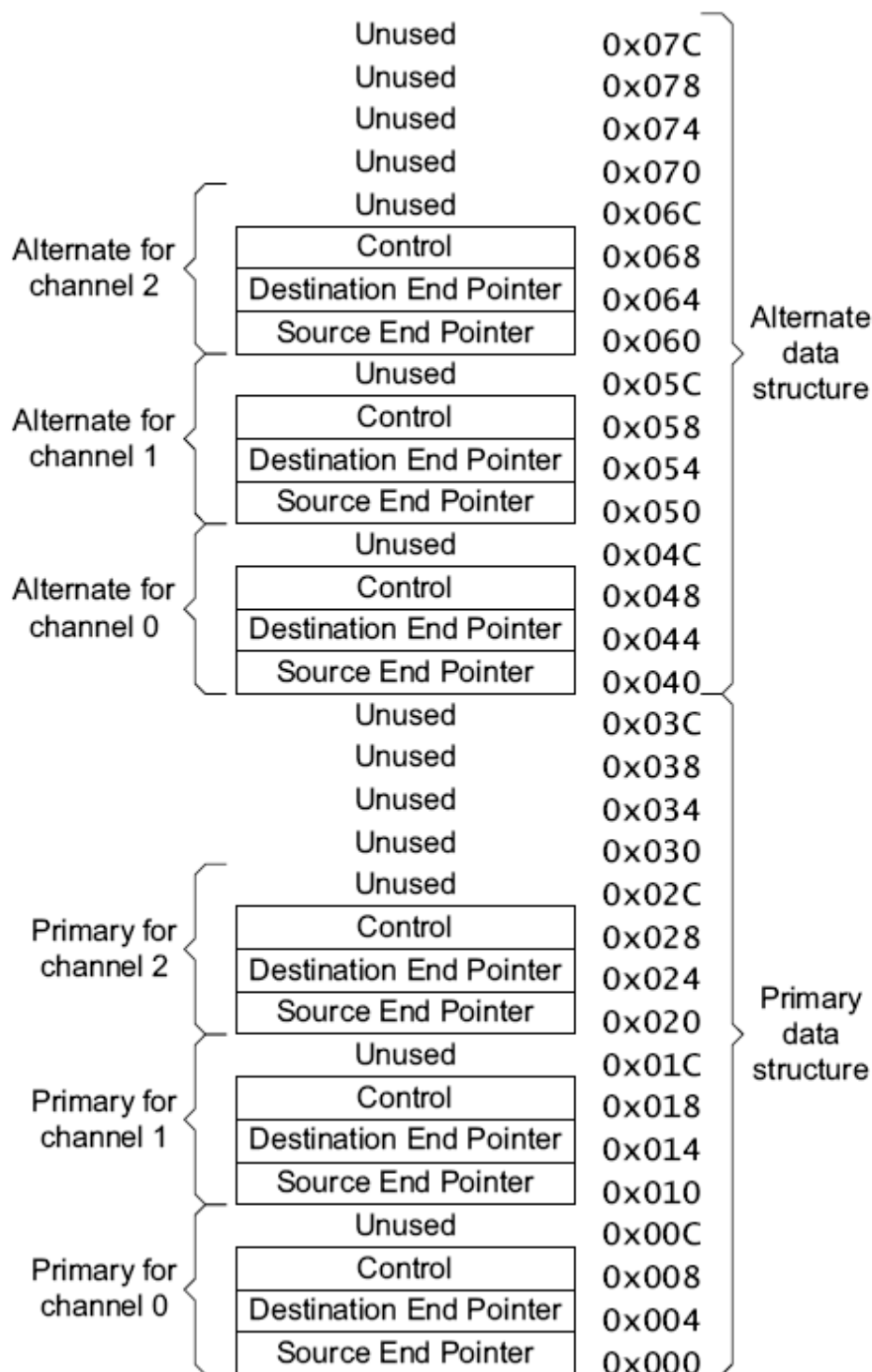


Рисунок 93 – Карта памяти для трех каналов DMA, включая альтернативную структуру управляющих данных (где Destination end pointer – указатель конца данных приемника; Source end pointer – указатель конца данных источника; Control – управление)

Пример структуры управляющих данных использует 128 байт системной памяти. В этом примере контроллер использует младшие 6 разрядов адреса для доступа ко всем элементам структуры управляющих данных, и поэтому базовый адрес структуры должен быть 0xXXXXXX00, далее 0xXXXXXX80.

Таблица 406 перечисляет все разрешенные значения базового адреса для первичной структуры управляющих данных, в зависимости от количества каналов DMA, использованных в контроллере.

Таблица 406 – Разрешенные базовые адреса

Кол-во каналов DMA	Разрешенные значения базового адреса для первичной структуры управляющих данных
1	0xXXXXXX00, 0xXXXXXX20, 0xXXXXXX40, 0xXXXXXX60, 0xXXXXXX80, 0xXXXXXXA0, 0xXXXXXXC0, 0xXXXXXXE0
2	0xXXXXXX00, 0xXXXXXX40, 0xXXXXXX80, 0xXXXXXXC0
3-4	0xXXXXXX00, 0xXXXXXX80
5-8	0xXXXXX000, 0xXXXXX100, 0xXXXXX200, 0xXXXXX300, 0xXXXXX400, 0xXXXXX500, 0xXXXXX600, 0xXXXXX700, 0xXXXXX800, 0xXXXXX900, 0xXXXXXA00, 0xXXXXXB00, 0xXXXXXC00, 0xXXXXXD00, 0xXXXXXE00, 0xXXXXXF00,
9-16	0xXXXXX000, 0xXXXXX200, 0xXXXXX400, 0xXXXXX600, 0xXXXXX800, 0xXXXXXA00, 0xXXXXXC00, 0xXXXXXE00
17-32	0xXXXXX000, 0xXXXXX400, 0xXXXXX800, 0xXXXXXC00

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

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

Указатель конца данных источника

Область памяти под названием src_data_end_ptr содержит указатель на последний адрес месторасположения данных источника. Таблица 407 перечисляет значения разрядов этой области.

Таблица 407 – Значения разрядов src_data_end_ptr

Разряд	Имя	Описание
31...0	src_data_end_ptr	указатель на последний адрес данных источника

Перед тем как контроллер выполнит передачу DMA, необходимо определить эту область памяти. Контроллер считывает значение этой области перед началом 2^R передачи DMA.

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

Указатель конца данных приемника

Область памяти под названием `dst_data_end_ptr` содержит указатель на последний адрес месторасположения данных приемника. Таблица 408 перечисляет значения разрядов этой области.

Таблица 408 – Значения разрядов `dst_data_end_ptr`

Разряд	Имя	Описание
31...0	<code>dst_data_end_ptr</code>	указатель на последний адрес данных приемника

Перед тем как контроллер выполнит передачу DMA, необходимо определить эту область памяти. Контроллер считывает значение этой области перед началом 2^R передачи DMA.

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

Разряды управления

Область памяти под названием `channel_cfg` обеспечивает управление каждой передачей DMA. Таблица 409 показывает название разрядов этой области.

Таблица 409 – Название разрядов области памяти `channel_cfg`

Номер	31...30	29...28	27...26	25...24	23...21
Доступ	R/W	R/W	R/W	R/W	R/W
Сброс	10	10	10	10	-
	<code>dst_inc</code>	<code>dst_size</code>	<code>src_inc</code>	<code>src_size</code>	<code>dst_prot_ctrl</code>

Номер	20...18	17...14	13...4	3	2...0
Доступ	R/W	R/W	R/W	R/W	R/W
Сброс	-	0010	-	0	100
	<code>Src_prot_ctrl</code>	<code>R_power</code>	<code>n_minus_1</code>	<code>next_useburst</code>	<code>cycle_ctrl</code>

Таблица 410 – Назначение разрядов `channel_cfg`

Разряд	Имя	Описание
31...30	<code>dst_inc</code>	Шаг инкремента адреса приемника Шаг инкремента адреса зависит от разрядности данных источника Разрядность данных источника = байт 2'b00 = байт; 2'b01 = полуслово (16-разрядное слово); 2'b10 = слово (32-разрядное слово); 2'b11 = нет инкремента. Адрес остается равным значению области памяти <code>dst_data_end_ptr</code> . Разрядность данных источника = полуслово 2'b00 = зарезервировано; 2'b01 = полуслово; 2'b10 = слово; 2'b11 = нет инкремента. Адрес остается равным значению области памяти <code>dst_data_end_ptr</code> . Разрядность данных источника = слово 2'b00 = зарезервировано; 2'b01 = зарезервировано;

Разряд	Имя	Описание
		2'b10 = слово; 2'b11 = нет инкремента. Адрес остается равным значению области памяти <code>dst_data_end_ptr</code>.
29...28	<code>dst_size</code>	Размерность данных приемника Примечание — Значение этого поля должно быть равно значению поля <code>src_size</code>.
27...26	<code>src_inc</code>	Шаг инкремента адреса источника Шаг инкремента адреса зависит от разрядности данных источника Разрядность данных источника = байт 2'b00 = байт; 2'b01 = полуслово (16-разрядное слово); 2'b10 = слово (32-разрядное слово); 2'b11 = нет инкремента. Адрес остается равным значению области памяти <code>src_data_end_ptr</code>. Разрядность данных источника = полуслово 2'b00 = зарезервировано; 2'b01 = полуслово; 2'b10 = слово; 2'b11 = нет инкремента. Адрес остается равным значению области памяти <code>src_data_end_ptr</code>. Разрядность данных источника = слово 2'b00 = зарезервировано; 2'b01 = зарезервировано; 2'b10 = слово; 2'b11 = нет инкремента. Адрес остается равным значению области памяти <code>src_data_end_ptr</code>.
25...24	<code>src_size</code>	Задаёт размерность данных источника 2'b00 = байт; 2'b01 = полуслово (16-разрядное слово); 2'b10 = слово (32-разрядное слово); 2'b11 = зарезервировано.
23...21	<code>dst_prot_ctrl</code>	Задаёт состояние <code>HPROT[3:1]</code>, когда контроллер записывает данные в приемник. Разряд [23] управляет разрядом <code>HPROT[3]</code>: 0 = <code>HPROT[3]</code> в состоянии 0 и доступ не кэшируется; 1 = <code>HPROT[3]</code> в состоянии 1 и доступ кэшируется. Разряд [22] управляет разрядом <code>HPROT[2]</code>: 0 = <code>HPROT[2]</code> в состоянии 0 и доступ не буферизуется; 1 = <code>HPROT[2]</code> в состоянии 1 и доступ буферизуется. Разряд [21] управляет разрядом <code>HPROT[1]</code>: 0 = <code>HPROT[1]</code> в состоянии 0 и доступ не привилегированный; 1 = <code>HPROT[1]</code> в состоянии 1 и доступ привилегированный.
20...18	<code>src_prot_ctrl</code>	Задаёт состояние <code>HPROT[3:1]</code>, когда контроллер считывает данные из источника. Разряд [20] управляет разрядом <code>HPROT[3]</code>: 0 = <code>HPROT[3]</code> в состоянии 0 и доступ не кэшируется; 1 = <code>HPROT[3]</code> в состоянии 1 и доступ кэшируется. Разряд [19] управляет разрядом <code>HPROT[2]</code>: 0 = <code>HPROT[2]</code> в состоянии 0 и доступ не буферизуется; 1 = <code>HPROT[2]</code> в состоянии 1 и доступ буферизуется. Разряд [18] управляет разрядом <code>HPROT[1]</code>: 0 = <code>HPROT[1]</code> в состоянии 0 и доступ не привилегированный; 1 = <code>HPROT[1]</code> в состоянии 1 и доступ привилегированный.
17...14	<code>R_power</code>	Задаёт количество передач DMA до выполнения контроллером процедуры арбитража. Возможные значения: 4'b0000 - арбитраж производится после каждой передачи DMA; 4'b0001 – арбитраж производится после 2 передач DMA;

Разряд	Имя	Описание
		4'b0010 – арбитраж производится после 4 передач DMA; 4'b0011 – арбитраж производится после 8 передач DMA; 4'b0100 – арбитраж производится после 16 передач DMA; 4'b0101 – арбитраж производится после 32 передач DMA; 4'b0110 – арбитраж производится после 64 передач DMA; 4'b0111 – арбитраж производится после 128 передач DMA; 4'b1000 – арбитраж производится после 256 передач DMA; 4'b1001 – арбитраж производится после 512 передач DMA; 4'b1010 – 4'b1111 – арбитраж производится после 1024 передач DMA. Это означает, что арбитраж не производится, так как максимальное количество передач DMA равно 1024
13...4	n_minus_1	Перед выполнением цикла DMA эти разряды указывают общее количество передач DMA, из которых состоит цикл DMA. Необходимо установить эти разряды в значение, соответствующее размеру желаемого цикла DMA. 10-разрядное число плюс 1 задает количество передач DMA. Возможные значения: 10'b0000000000 = 1 передача DMA; 10'b0000000001 = 2 передачи DMA; 10'b0000000010 = 3 передачи DMA; 10'b0000000011 = 4 передачи DMA; 10'b0000000100 = 5 передач DMA; 10'b0000000101 = 6 передач DMA; 10'b1111111111 = 1024 передачи DMA. Контроллер обновит это поле перед тем, как произвести процесс арбитража. Это позволяет контроллеру хранить количество оставшихся передач DMA до завершения цикла DMA
3	next_useburst	Контролирует, не установлен ли chnl_useburst_set[C] в состояние 1, если контроллер работает в режиме работы с периферией «Исполнение с изменением конфигурации», и, если контроллер завершает цикл DMA, используя альтернативные управляющие данные. Примечание – Перед завершением цикла DMA, использующего альтернативные управляющие данные, контроллер устанавливает chnl_useburst_set[C] в значение 0, если количество оставшихся передач DMA меньше, чем 2^R. Установка next_useburst разряда определяет, будет ли контроллер дополнительно переопределять разряд chnl_useburst_set[C]. Если контроллер выполняет цикл DMA в режиме работы с периферией «Исполнение с изменением конфигурации», то после окончания цикла, использующего альтернативные управляющие данные, происходит следующее в зависимости от состояния next_useburst: 0 – контроллер не изменяет значение chnl_useburst_set[C]. Если chnl_useburst_set[C] установлен в 0, то для всех оставшихся циклов DMA в режиме работы с периферией «Исполнение с изменением конфигурации», контроллер отвечает на запросы по dma_req[] и dma_sreq[], при выполнении циклов DMA он использует альтернативные управляющие данные. 1 – контроллер изменяет значение chnl_useburst_set[C] в состояние 1. Поэтому для оставшихся циклов DMA в режиме работы с периферией «Исполнение с изменением конфигурации», контроллер реагирует только на запросы по dma_req[], при выполнении циклов DMA он использует альтернативные управляющие данные.
2...0	cycle_ctrl	Режим работы при выполнении цикла DMA: 3'b000 Стоп. Означает, что структура управляющих данных является «неправильной»;

Разряд	Имя	Описание
		3'b001 Основной. Контроллер должен получить новый запрос для окончания цикла DMA, перед этим он должен выполнить процедуру арбитража; 3'b010 Авто-запрос. Контроллер автоматически осуществляет запрос на обработку по соответствующему каналу в течение процедуры арбитража. Это означает, что начального запроса на обработку достаточно для выполнения цикла DMA; 3'b011 Пинг-понг. Контроллер выполняет цикл DMA используя одну из структур управляющих данных. По окончании выполнения цикла DMA, контроллер; выполняет следующий цикл DMA, используя другую структуру. Контроллер сигнализирует об окончании каждого цикла DMA, позволяя процессору перенастраивать неактивную структуру данных. Контроллер продолжает выполнять циклы DMA, до тех пор, пока он не прочитает «неправильную» структуру данных или пока процессор не изменит cycle_ctrl поле в состояние 3'b001 или 3'b010; 3'b100 Режим работы с памятью «Исполнение с изменением конфигурации». Смотрите соответствующий раздел. При работе контроллера в данном режиме значение этого поля в первичной структуре управляющих данных должно быть 3'b100; 3'b101 Режим работы с памятью «Исполнение с изменением конфигурации». Смотрите соответствующий раздел. При работе контроллера в данном режиме значение этого поля в альтернативной структуре управляющих данных должно быть 3'b101; 3'b110 Режим работы с периферией «исполнение с изменением конфигурации». Смотрите соответствующий раздел. При работе контроллера в данном режиме значение этого поля в первичной структуре управляющих данных должно быть 3'b110; 3'b111 Режим работы с периферией «исполнение с изменением конфигурации». Смотрите соответствующий раздел. При работе контроллера в данном режиме значение этого поля в альтернативной структуре управляющих данных должно быть 3'b111

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

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

После выполнения контроллером N передач, контроллер устанавливает значение поля cycle_ctrl в 3'b000, делая тем самым channel_cfg данные «неправильными». Это позволяет избежать повторения выполненной передачи DMA.

Вычисление адреса

Для вычисления адреса источника передачи DMA, контроллер выполняет сдвиг влево значения n_minus_1 на количество разрядов, соответствующее полю src_inc , и затем вычитает получившееся значение от значения указателя адреса конца данных источника. Подобным образом вычисляется адрес передатчика передачи DMA, контроллер выполняет сдвиг влево значения n_minus_1 на количество разрядов, соответствующее полю dst_inc , и затем вычитает получившееся значение от значения указателя адреса конца данных приемника.

В зависимости от значения полей src_inc и dst_inc вычисления адресов приемника и источника выполняются по следующим уравнениям:

$src_inc=b00$ and $dst_inc=b00$

- адрес источника = $src_data_end_ptr - n_minus_1$
- адрес приемника = $dst_data_end_ptr - n_minus_1$.

$src_inc=b01$ and $dst_inc=b01$

- адрес источника = $src_data_end_ptr - (n_minus_1 \ll 1)$
- адрес приемника = $dst_data_end_ptr - (n_minus_1 \ll 1)$.

$src_inc=b01$ and $dst_inc=b10$

- адрес источника = $src_data_end_ptr - (n_minus_1 \ll 2)$
- адрес приемника = $dst_data_end_ptr - (n_minus_1 \ll 2)$.

$src_inc=b11$ and $dst_inc=b11$

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

Таблица 411 перечисляет адреса приемника цикла DMA для 6 слов.

Таблица 411 – Цикла DMA для 6 слов с пословным инкрементом

Начальные значения channel_cfg перед циклом DMA				
$src_size=2'b10, dst_inc=2'b10, n_minus_1=3'b101, cycle_ctrl=1$				
DMA передачи	Указатель конца данных	Счетчик	Отличие*	Адрес
	0x2AC	5	0x14	0x298
	0x2AC	4	0x10	0x29C
	0x2AC	3	0xC	0x2A0
	0x2AC	2	0x8	0x2A4
	0x2AC	1	0x4	0x2A8
	0x2AC	0	0x0	0x2AC
Конечные значения channel_cfg после цикла DMA				
$src_size=2'b10, dst_inc=2'b10, n_minus_1=0, cycle_ctrl=0$				

* это значение, полученное после сдвига влево значения счетчика на количество разрядов соответствующее dst_inc .

Таблица 412 перечисляет адреса приемника для передач DMA 12 байт с использованием «полусловного» инкремента.

Таблица 412 – Цикла DMA для 12 байт с «полусловным» инкрементом

Начальные значения channel_cfg перед циклом DMA				
src_size=2'b00, dst_inc=2'b01, n_minus_1=4'b1011, cycle_ctrl=1, R_power=2'b11				
DMA передачи	Указатель конца данных	Счетчик	Отличие*	Адрес
	0x5E7	11	0x16	0x5D1
	0x5E7	10	0x14	0x5D3
	0x5E7	9	0x12	0x5D5
	0x5E7	8	0x10	0x5D7
	0x5E7	7	0xE	0x5D9
	0x5E7	6	0xC	0x5DB
	0x5E7	5	0xA	0x5DD
	0x5E7	4	0x8	0x5DF
Значения channel_cfg после 2 ^R передач DMA				
src_size=2'b00, dst_inc=2'b01, n_minus_1=3'b011, cycle_ctrl=1, R_power=2'b11				
DMA передачи	0x5E7	3	0x6	0x5E1
	0x5E7	2	0x4	0x5E3
	0x5E7	1	0x2	0x5E5
	0x5E7	0	0x0	0x5E7
Конечные значения channel_cfg после цикла DMA				
src_size=2'b00, dst_inc=2'b01, n_minus_1=0, cycle_ctrl=0**, R_power=2'b11				

* это значение, полученное после сдвига влево значения счетчика на количество разрядов, соответствующее dst_inc.

** после окончания цикла DMA контроллер делает channel_cfg «неправильным», сбрасывая в 0 поле cycle_ctrl.

1.6 Описание регистров контроллера DMA

Данный раздел описывает регистры контроллера и управление контроллером через них.

Раздел содержит следующие сведения:

- о регистровой модели контроллера;
- описание регистров.

Основные положения регистровой модели контроллера:

- нужно избегать адресации при доступе к зарезервированным или неиспользованным адресам, так как это может привести к непредсказуемым результатам;
- необходимо заполнять неиспользуемые или зарезервированные разряды регистров нулями при записи и игнорировать значения таких разрядов при считывании, кроме случаев, специально описанных в разделе;
- системный сброс или сброс по установке питания сбрасывает все регистры в состояние 0, кроме случаев, специально описанных в разделе;

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

Таблица 413 – Перечень регистров контроллера

Наименование	Смещение относительно базового адреса	Тип	Значение по сбросу	Описание
status	0x000	RO	0x-nn0000*	Статусный регистр DMA
cfg	0x004	WO	-	Регистр конфигурации DMA
ctrl_base_ptr	0x008	R/W	0x00000000	Регистр базового адреса управляющих данных каналов
alt_ctrl_base_ptr	0x00C	RO	0x000000nn**	Регистр базового адреса альтернативных управляющих данных каналов
waitonreq_status	0x010	RO	0x00000000	Регистр статуса ожидания запроса на обработку каналов
chnl_sw_request	0x014	WO	-	Регистр программного запроса на обработку каналов
chnl_useburst_set	0x018	R/W	0x00000000	Регистр установки пакетного обмена каналов
chnl_useburst_clr	0x01C	WO	-	Регистр сброса пакетного обмена каналов
chnl_req_mask_set	0x020	R/W	0x00000000	Регистр маскирования запросов на обслуживание каналов
chnl_req_mask_clr	0x024	WO	-	Регистр очистки маскирования запросов на обслуживание каналов
chnl_enable_set	0x028	R/W	0x00000000	Регистр установки разрешения каналов
chnl_enable_clr	0x02C	WO	-	Регистр сброса разрешения каналов
chnl_pri_alt_set	0x030	R/W	0x00000000	Регистр установки первичной/альтернативной структуры управляющих данных каналов
chnl_pri_alt_clr	0x034	WO	-	Регистр сброса первичной/альтернативной структуры управляющих данных каналов
chnl_priority_set	0x038	R/W	0x00000000	Регистр установки приоритета каналов
chnl_priority_clr	0x03C	WO	-	Регистр сброса приоритета каналов
-	0x040-0x048	-	-	зарезервировано
err_clr	0x04C	R/W	0x00000000	Регистр сброса флага ошибки
-	0x050-0xDFC	-	-	зарезервировано

* – значение по сбросу зависит от количества каналов DMA, использованных в контроллере, а также от наличия интегрированной схемы тестирования.

** – значение по сбросу зависит от количества каналов DMA, использованных в контроллере.

1.6.1 Статусный регистр DMA STATUS

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

Таблица 414 – Статусный регистр DMA

Номер	31...28	27...21	20...16	15...8	7...4	3...1	0
Доступ	RO	U	RO	U	RO	U	RO
Сброс	0	0	0	0	0	0	0

	test_status	-	chnls_minus1	-	state	-	master_enable
--	-------------	---	--------------	---	-------	---	---------------

Таблица 415 – Назначение разрядов регистра dma_status

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...28	test_status	Значение при чтении: 4'b0000 = контроллер не имеет интегрированной схемы тестирования; 4'b0001 = контроллер имеет интегрированную схему тестирования; 4'b0010 – 4'b1111 = не определено
27...21	-	Не определено
20...16	chnls_minus1	Количество доступных каналов DMA минус 1. Например: 5'b00000 = контроллер имеет 1 канал DMA; 5'b00001 = контроллер имеет 2 канала DMA; 5'b00010 = контроллер имеет 3 канала DMA; ... 5'b11111 = контроллер имеет 32 канала DMA
15...8	-	Не определено
7...4	state	Текущее состояние автомата управления контроллера. Состояние может быть одним из следующих: 4'b0000 = в покое; 4'b0001 = чтение управляющих данных канала; 4'b0010 = чтение указателя конца данных источника; 4'b0011 = чтение указателя конца данных приемника; 4'b0100 = чтение данных источника; 4'b0101 = запись данных в приемник; 4'b0110 = ожидание запроса на выполнение DMA; 4'b0111 = запись управляющих данных канала; 4'b1000 = приостановлен; 4'b1001 = выполнен; 4'b1010 = режим работы с периферией «Исполнение с изменением конфигурации»; 4'b1011 – 4'b1111 = не определено
3...1	-	Не определено
0	master_enable	Состояние контроллера: 0 = работа контроллера запрещена; 1 = работа контроллера разрешена

1.6.2 Регистр конфигурации DMA CFG

Данный регистр имеет доступ только на запись. Регистр определяет состояние контроллера.

Таблица 416 – Регистр конфигурации DMA

Номер	31...8	7...5	4...1	0
Доступ	U	WO	U	WO
Сброс	0	0	0	0
	-	chnl_prot_ctrl	-	master_enable

Таблица 417 – Назначение разрядов регистра dma_cfg

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
--------	-------------------------	--

31...8	-	Не определено, следует записывать 0
7...5	chnl_prot_ctrl	Определяет уровни индикации сигналов HPROT[3:1] защиты шины AHB-Lite: Разряд 7 управляет сигналом HPROT[3], с целью индикации о появлении доступа с кэшированием; Разряд 6 управляет сигналом HPROT[2], с целью индикации о появлении доступа с буферизацией; Разряд 5 управляет сигналом HPROT[1], с целью индикации о появлении привилегированного доступа. Примечание — Если разряд[n] = 1, то соответствующий сигнал HPROT в состоянии 1. Если разряд[n] = 0, то соответствующий сигнал HPROT в состоянии 0
4...1	-	Не определено. Следует записывать 0
0	master_enable	Определяет состояние контроллера: 0 – запрещает работу контроллера; 1 – разрешает работу контроллера

1.6.3 Регистр базового адреса управляющих данных каналов CTRL_BASE_PTR

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

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

Размер системной памяти, предназначенной контроллеру, зависит от количества каналов DMA, использующихся контроллером, а также от возможности использования альтернативных управляющих данных каналов. Поэтому количество разрядов регистра, необходимых для задания базового адреса, варьируется и зависит от варианта построения системы.

Если контроллер находится в состоянии сброса, то чтение регистра запрещено.

Таблица 418 – Регистр базового адреса управляющих данных каналов

Номер	31...10	9...0
Доступ	R/W	U
Сброс	0	0
	ctrl_base_ptr	-

Таблица 419 – Назначение разрядов регистра ctrl_base_ptr

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...10	ctrl_base_ptr	Указатель на базовый адрес первичной структуры управляющих данных. См. соответствующий раздел
9...0	-	Не определено. Следует записывать 0

1.6.4 Регистр базового адреса альтернативных управляющих данных каналов ALT_CTRL_BASE_PTR

Данный регистр имеет доступ только на чтение. Регистр возвращает при чтении указатель базового адреса альтернативных управляющих данных каналов. Если контроллер

находится в состоянии сброса, то чтение регистра запрещено. Этот регистр позволяет не производить вычисления базового адреса альтернативных управляющих данных каналов.

Таблица 420 – Регистр базового адреса альтернативных управляющих данных каналов

Номер	31... 0
Доступ	RO
Сброс	0
	Alt_ctrl_base_ptr

Таблица 421 – Назначение разрядов регистра alt_ctrl_base_ptr

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	alt_ctrl_base_ptr	Указатель базового адреса альтернативной структуры управляющих данных

1.6.5 Регистр статуса ожидания запроса на обработку каналов WAITONREQ_STATUS

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

Таблица 422 – Регистр статуса ожидания запроса на обработку каналов

Номер	31		2	1	0
Доступ	RO		RO	RO	RO
Сброс	0		0	0	0
	dma_waitonreq_status for dma_waitnreg [31]		dma_waitonreq_status for dma_waitnreg [2]	dma_waitonreq_status for dma_waitnreg [1]	dma_waitonreq_status for dma_waitnreg [0]

Таблица 423 – Назначение разрядов регистра dma_waitonreq_status

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	dma_waitonreq_status	Состояние сигналов ожидания запроса на обработку каналов DMA. При чтении: Разряд [C] = 1 означает, что dma_waitonreq[C] в состоянии 1; Разряд [C] = 0 означает, что dma_waitonreq[C] в состоянии 0.

1.6.6 Регистр программного запроса на обработку каналов CHNL_SW_REQUEST

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

Таблица 424 – Регистр программного запроса на обработку каналов

Номер	31		2	1	0
Доступ	WO		WO	WO	WO
Сброс	0		0	0	0
	chnl_sw_request for channel [31]		chnl_sw_request for channel [2]	chnl_sw_request for channel [1]	chnl_sw_request for channel [0]

Таблица 425 – Назначение разрядов регистра chnl_sw_request

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	chnl_sw_request	Устанавливает соответствующий разряд для генерации программного запроса на выполнение цикла DMA по соответствующему каналу DMA. При записи: Разряд [C] = 0 означает, что запрос на выполнение цикла DMA по каналу C не будет установлен; Разряд [C] = 1 означает, что запрос на выполнение цикла DMA по каналу C будет установлен. Запись разряда, соответствующего нереализованному каналу, означает, что запрос на выполнение цикла DMA не будет установлен.

1.6.7 Регистр установки пакетного обмена каналов CHNL_USEBURST_SET

Данный регистр имеет доступ на чтение и запись. Регистр отключает выполнение одиночных запросов по установке dma_sreq[] и поэтому будут обрабатываться и исполняться только запросы по dma_req[]. Регистр возвращает при чтении состояние установок пакетного обмена

Таблица 426 – Регистр установки пакетного обмена каналов

Номер	31		2	1	0
Доступ	R/W		R/W	R/W	R/W
Сброс	0		0	0	0
	chnl_useburst_set for channel [31]		chnl_useburst_set for channel [2]	chnl_useburst_set for channel [1]	chnl_useburst_set for channel [0]

Таблица 427 – Назначение разрядов регистра chnl_useburst_set

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	chnl_useburst_set	Отключает обработку запросов на выполнение циклов DMA от dma_sreq[] и возвращает при чтении состоянии этих

		настроек. При чтении: Разряд [C] = 0 означает, что канал DMA C выполняет циклы DMA в ответ на запросы, полученные от dma_sreq[] и dma_req[]. Контроллер выполняет одиночные передачи или 2 ^R передач. Разряд [C] = 1 означает, что канал DMA C выполняет циклы DMA в ответ на запросы, полученные только от dma_req[]. Контроллер выполняет 2 ^R передач. При записи: Разряд [C] = 0 не дает эффекта. Необходимо использовать chnl_useburst_clr регистр и установить соответствующий разряд C в 0; Разряд [C] = 1 отключает возможность обрабатывать запросы на выполнение циклов DMA, полученные от dma_sreq[]. Контроллер выполняет 2 ^R передач. Запись разряда, соответствующего нереализованному каналу, не дает никакого эффекта
--	--	--

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

Примечание – При программировании channel_cfg значением N меньшим, чем 2^R, запрещена установка соответствующего разряда chnl_useburst_set в случае, если периферийный блок не поддерживает сигнал dma_req[].

В режиме работы с периферией «исполнение с изменением конфигурации», если разряд next_useburst установлен в channel_cfg, то контроллер устанавливает chnl_useburst_set [C] в 1 после окончания цикла DMA, использующего альтернативные управляющие данные.

1.6.8 Регистр сброса пакетного обмена каналов CHNL_USEBURST_CLR

Данный регистр имеет доступ только на запись. Регистр разрешает выполнение одиночных запросов по установке dma_sreq[].

Таблица 428 – Регистр сброса пакетного обмена каналов

Номер	31		2	1	0
Доступ	WO		WO	WO	WO
Сброс	0		0	0	0
	chnl_useburst_clr for channel [31]		chnl_useburst_clr for channel [2]	chnl_useburst_clr for channel [1]	chnl_useburst_clr for channel [0]

Таблица 429 – Назначение разрядов регистра chnl_useburst_clr

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
--------	-------------------------	--

31...0	chnl_useburst_clr	Установка соответствующего разряда разрешает обработку запросов на выполнение циклов DMA от dma_sreq[]. При записи: Разряд [C] = 0 не дает эффекта. Необходимо использовать chnl_useburst_set регистр для отключения обработки запросов от dma_sreq[]; Разряд [C] = 1 разрешает обрабатывать запросы на выполнение циклов DMA, полученные от dma_sreq[]. Запись разряда, соответствующего нереализованному каналу, не дает никакого эффекта.
--------	-------------------	--

1.6.9 Регистр маскирования запросов на обслуживание каналов CHNL_REQ_MASK_SET

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

Таблица 430 – Регистр маскирования запросов на обслуживание каналов

Номер	31		2	1	0
Доступ	R/W		R/W	R/W	R/W
Сброс	0		0	0	0
	chnl_reg_mask_set for dma_reg [31] and dma_sreg [31]		chnl_reg_mask_set for dma_reg [2] and dma_sreg [2]	chnl_reg_mask_set for dma_reg [1] and dma_sreg [1]	chnl_reg_mask_set for dma_reg [0] and dma_sreg [0]

Таблица 431 – Назначение разрядов регистра chnl_req_mask_set

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	chnl_req_mask_set	Отключает обработку запросов по dma_sreq[] и dma_req[] на выполнение циклов DMA от каналов и возвращает при чтении состоянии этих настроек. При чтении: Разряд [C] = 0 означает, что канал DMA C выполняет циклы DMA в ответ на поступающие запросы; Разряд [C] = 1 означает, что канал DMA C не выполняет циклы DMA в ответ на поступающие запросы. При записи: Разряд [C] = 0 не дает эффекта. Необходимо использовать chnl_req_mask_clr регистр для разрешения установки запросов; Разряд [C] = 1 отключает установку запросов на выполнение циклов DMA, по dma_sreq[] и dma_req[]. Запись разряда, соответствующего нереализованному каналу, не дает никакого эффекта.

1.6.10 Регистр очистки маскирования запросов на обслуживание каналов CHNL_REQ_MASK_CLR

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

Таблица 432 – Регистр очистки маскирования запросов на обслуживание каналов

Номер	31		2	1	0
Доступ	WO		WO	WO	WO
Сброс	0		0	0	0
	chnl_req_mask_clr for dma_reg [31] and dma_sreg [31]		chnl_req_mask_clr for dma_reg [2] and dma_sreg [2]	chnl_req_mask_clr for dma_reg [1] and dma_sreg [1]	chnl_req_mask_clr for dma_reg [0] and dma_sreg [0]

Таблица 433 – Назначение разрядов регистра chnl_req_mask_clr

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	chnl_req_mask_clr	Установка соответствующего разряда разрешает установку запросов по dma_sreq[] и dma_req[] на выполнение циклов DMA от каналов. При записи: Разряд [C] = 0 не дает эффекта. Необходимо использовать chnl_req_mask_set регистр для отключения установки запросов; Разряд [C] = 1 разрешает установку запросов на выполнение циклов DMA, по dma_sreq[] и dma_req[]. Запись разряда, соответствующего нереализованному каналу, не дает никакого эффекта.

1.6.11 Регистр установки разрешения каналов CHNL_ENABLE_SET

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

Таблица 434 – Регистр установки разрешения каналов

Номер	31		2	1	0
Доступ	R/W		R/W	R/W	R/W
Сброс	0		0	0	0

	chnl_enable_set for channel [31]		chnl_enable_set for channel [2]	chnl_enable_set for channel [1]	chnl_enable_set for channel [0]
--	-------------------------------------	-------	------------------------------------	------------------------------------	------------------------------------

Таблица 435 – Назначение разрядов регистра chnl_enable_set

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	chnl_enable_set	Разрешает работу каналов DMA и возвращает при чтении состояние этих настроек. При чтении: Разряд [C] = 0 означает, что канал DMA C отключен; Разряд [C] = 1 означает, что работа канала DMA C разрешена. При записи: Разряд [C] = 0 не дает эффекта. Необходимо использовать chnl_enable_clr регистр для отключения канала; Разряд [C] = 1 разрешает работу канала DMA C. Запись разряда, соответствующего нереализованному каналу, не дает никакого эффекта.

1.6.12 Регистр сброса разрешения каналов CHNL_ENABLE_CLR

Данный регистр имеет доступ только на запись. Регистр запрещает работу каналов DMA.

Таблица 436 – Регистр сброса разрешения каналов

Номер	31		2	1	0
Доступ	WO		WO	WO	WO
Сброс	0		0	0	0
	chnl_enable_clr for channel 31		chnl_enable_clr for channel 2	chnl_enable_clr for channel 1	chnl_enable_clr for channel 0

Таблица 437 – Назначение разрядов регистра chnl_enable_clr

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	chnl_enable_clr	Установка соответствующего разряда запрещает работу соответствующего канала DMA. При записи: Разряд [C] = 0 не дает эффекта. Необходимо использовать chnl_enable_set регистр для разрешения работы канала; Разряд [C] = 1 запрещает работу канала DMA C. Запись разряда, соответствующего нереализованному каналу, не дает никакого

		эффекта. Примечание – Контроллер может отключить канал DMA, установив соответствующий разряд в следующих случаях: - при завершении цикла DMA; - при чтении из channel_cfg с полем cycle_ctrl установленным в 3'b000; - при появлении ошибки на шине AHB-Lite.
--	--	---

1.6.13 Регистр установки первичной/альтернативной структуры управляющих данных каналов CHNL_PRI_ALT_SET

Данный регистр имеет доступ на запись и чтение. Регистр разрешает работу канала DMA с использованием альтернативной структуры управляющих данных. Чтение регистра возвращает состояние каналов DMA (какую структуру управляющих данных использует каждый канал DMA).

Таблица 438 – Регистр установки первичной/альтернативной структуры управляющих данных каналов

Номер	31		2	1	0
Доступ	R/W		R/W	R/W	R/W
Сброс	0		0	0	0
	chnl_pri_alt_set for channel [31]		chnl_pri_alt_set for channel [2]	chnl_pri_alt_set for channel [1]	chnl_pri_alt_set for channel [0]

Таблица 439 – Назначение разрядов регистра chnl_pri_alt_set

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
--------	-------------------------	--

31...0	chnl_pri_alt_set	Установка соответствующего разряда подключает использование альтернативных управляющих данных для соответствующего канала DMA, чтение возвращает состояние этих настроек. При чтении: Разряд [C] = 0 означает, что канал DMA C использует первичную структуру управляющих данных; Разряд [C] = 1 означает, что канал DMA C использует альтернативную структуру управляющих данных. При записи: Разряд [C] = 0 не дает эффекта. Необходимо использовать chnl_pri_alt_clr регистр для сброса разряда [C] в 0; Разряд [C] = 1 подключает использование альтернативной структуры управляющих данных каналом DMA C. Запись разряда, соответствующего нереализованному каналу, не дает никакого эффекта. Примечание – Контроллер может переключить значение разряда chnl_pri_alt_set[C] в следующих случаях: - при завершении 4-х передач DMA указанных в первичной структуре управляющих данных при выполнении цикла DMA в режимах работы с памятью или периферией «исполнение с изменением конфигурации»; - при завершении всех передач DMA указанных в первичной структуре управляющих данных при выполнении цикла DMA в режиме «Пинг-понг»; - при завершении всех передач DMA указанных в альтернативной структуре управляющих данных при выполнении цикла DMA в режимах: - «пинг-понг»; - работа с памятью «Исполнение с изменением конфигурации»; - работа с периферией «Исполнение с изменением конфигурации»;
--------	------------------	--

1.6.14 Регистр сброса первичной/альтернативной структуры управляющих данных каналов CHNL_PRI_ALT_CLR

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

Таблица 440 – Регистр сброса первичной/альтернативной структуры управляющих данных каналов

Номер	31		2	1	0
Доступ	WO		WO	WO	WO
Сброс	0		0	0	0
	chnl_pri_alt_clr for channel [31]		chnl_pri_alt_clr for channel [2]	chnl_pri_alt_clr for channel [1]	chnl_pri_alt_clr for channel [0]

Таблица 441 – Назначение разрядов регистра chnl_pri_alt_clr

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
--------	-------------------------	--

31...0	chnl_pri_alt_clr	Установка соответствующего разряда подключает использование первичных управляющих данных для соответствующего канала DMA. При записи: Разряд [C] = 0 не дает эффекта. Необходимо использовать chnl_pri_alt_set регистр для выбора альтернативных управляющих данных; Разряд [C] = 1 подключает использование первичной структуры управляющих данных каналом DMA C. Запись разряда, соответствующего нереализованному каналу, не дает никакого эффекта. Примечание – Контроллер может переключить значение разряда chnl_pri_alt_clr[C] в следующих случаях: - при завершении 4-х передач DMA указанных в первичной структуре управляющих данных при выполнении цикла DMA в режимах работы с памятью или периферией «исполнение с изменением конфигурации»; - при завершении всех передач DMA указанных в первичной структуре управляющих данных при выполнении цикла DMA в режиме «пинг-понг»; - при завершении всех передач DMA указанных в альтернативной структуре управляющих данных при выполнении цикла DMA в режимах: - «пинг-понг» - работа с памятью «Исполнение с изменением конфигурации» - работа с периферией «Исполнение с изменением конфигурации»
--------	------------------	---

1.6.15 Регистр установки приоритета каналов CHNL_PRIORITY_SET

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

Таблица 442 – Регистр установки приоритета каналов

Номер	31		2	1	0
Доступ	R/W		R/W	R/W	R/W
Сброс	0		0	0	0
	chnl_priorit_set for channel [31]		chnl_priority_set for channel [2]	chnl_priority_set for channel [1]	chnl_priority_set for channel [0]

Таблица 443 – Назначение разрядов регистра chnl_priority_set

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	chnl_priority_set	Установка высокого приоритета каналу DMA, чтение возвращает состояние

		приоритета каналов DMA. При чтении: Разряд [C] = 0 означает, что каналу DMA C присвоен уровень приоритета по умолчанию; Разряд [C] = 1 означает, что каналу DMA C присвоен высокий уровень приоритета. При записи: Разряд [C] = 0 не дает эффекта. Необходимо использовать chnl_priority_clr регистр для установки каналу C уровня приоритета по умолчанию; Разряд [C] = 1 устанавливает каналу DMA C высокий уровень приоритета. Запись разряда, соответствующего нереализованному каналу, не дает никакого эффекта;
--	--	---

1.6.16 Регистр сброса приоритета каналов CHNL_PRIORITY_CLR

Данный регистр имеет доступ только на запись. Регистр позволяет присвоить каналу DMA уровень приоритета по умолчанию.

Таблица 444 – Регистр сброса приоритета каналов

Номер	31		2	1	0
Доступ	WO		WO	WO	WO
Сброс	0		0	0	0
	chnl_prioirit_clr for channel [31]		chnl_priority_clr for channel [2]	chnl_priority_clr for channel [1]	chnl_priority_clr for channel [0]

Таблица 445 – Назначение разрядов регистра chnl_priority_clr

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	chnl_priority_clr	Установка разряда присваивает соответствующему каналу DMA уровень приоритета по умолчанию. При записи: Разряд [C] = 0 не дает эффекта. Необходимо использовать chnl_priority_set регистр для установки каналу C высокого уровня приоритета. Разряд [C] = 1 устанавливает каналу DMA C уровень приоритета по умолчанию. Запись разряда, соответствующего нереализованному каналу, не дает никакого эффекта.

1.6.17 Регистр сброса флага ошибки ERR_CLR

Данный регистр имеет доступ на запись и чтение. Регистр позволяет сбрасывать сигнал dma_err в 0. Чтение регистра возвращает состояние сигнала dma_err.

Таблица 446 – Регистр сброса флага ошибки

Номер	31...1	0
Доступ	U	R/W
Сброс	0	0
	-	err_clr

Таблица 447 – Назначение разрядов регистра err_clr

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...1	-	Не определено. Следует записывать 0
0	err_clr	Установка сигнала в состояние 0, чтение возвращает состояние сигнала (флага) dma_err. При чтении: Разряд [C] = 0 означает, что dma_err находится в состоянии 0; Разряд [C] = 1 означает, что dma_err находится в состоянии 1. При записи: Разряд [C] = 0 не дает эффекта. Состояние dma_err останется неизменным; Разряд [C] = 1 сбрасывает сигнал (флаг) dma_err в состояние 0. Примечание – При сбросе сигнала dma_err одновременно с появлением ошибки на шине АНВ-Lite, приоритет отдается ошибке, и, следовательно, значение регистра (и dma_err) останется неизменным (несброшенным).

Прерывания

Согласно спецификации привилегированной системы команд, RISC-V ядро должно обеспечивать поддержку обработки программных прерываний (machine software interrupt) и прерываний таймера (machine timer interrupt). За генерацию запросов (сигналов) этих прерываний отвечает локальный контроллер прерываний (Core Local Interrupt Controller, CLINT). CLINT содержит регистры для управления программными прерываниями и прерываниями таймера. Карта регистров CLINT приведена в таблице ниже.

Регистры CLINT

Адрес	Название	Доступ
0x0200_0000	Регистр запроса/снятия программного прерывания	R/W
0x0200_4000	Регистр сравнения таймера	R/W
0x0200_BFF8	Регистр значения таймера	R

Регистр запроса/снятия прерывания

Биты	Название	Доступ	Описание	Значение
0	msip	R/W	Machine software interrupt pending	0

Регистр базового адреса вектора прерывания mtvec

Биты	Название	Доступ	Описание	Значение
31:2	BASE	R/W	Базовый адрес таблицы векторов прерываний. Должен быть предварительно инициализирован и выровнен по границе 4 байта.	0
1:0	MODE	R/W	0 – все прерывания устанавливают PC в значение BASE 1,2,3 – зарезервированные комбинации бит MODE.	0

Регистр номера события mcause

Биты	Название	Доступ	Описание	Значение
31	Interrupt	R/W	1 – в поле Exception Code номер события прерывания 0 – в поле Exception Code номер исключительной ситуации	0
30:0	Exception Code	R/W	Номер события прерывания или исключительной ситуации, где исключительная ситуация: 0 – инструкция по невыровненному адресу; 1- ошибка доступа инструкции; 2- нелегальная инструкция; 3 – точка останова; 4 – загрузка по невыровненному адресу; 5 – ошибка доступа при загрузке; 6 – сохранение по невыровненному адресу;	0

			7 – ошибка доступа при сохранении; 8 – environment вызывается из U-режима; 9 – environment вызывается из S-режима; 10 – зарезервировано; 11 – environment вызывается из M-режима; 12 – ошибка страницы инструкции; 13 – ошибка загрузки страницы; 14 – зарезервировано; 15 – ошибка сохранения страницы.	
--	--	--	--	--

Регистр значения таймера

Биты	Название	Доступ	Описание	Значение
31:0	mtime_lo	R	Младшая часть значения таймера	0
64:32	mtime_hi	R	Старшая часть значения таймера	0

Регистр содержит 64-битное значение таймера. Значение увеличивается по сигналу timer_pulse (частота LSE генератора 32 кГц) внешнего интерфейса процессорного комплекса.

Регистр сравнения таймера

Биты	Название	Доступ	Описание	Значение
31:0	mtimesmp_lo	R	Младшая часть значения регистра сравнения	0
64:32	mtimesmp_hi	R	Старшая часть значения регистра сравнения	0

Регистр содержит 64-битное значение сравнения с таймером. Запрос на прерывание таймера установлен, если значение таймера строго больше значения сравнения.

Периферийные блоки формируют прерывания с int_global0 по int_global29.

Таблица 448 Прерывания, формируемые периферийными блоками

Прерывания	Блок	Принцип формирования
int_global0	DMA	Прерывания от DMA DMA_ERR или DMA_DONE. Обработка прерываний от DMA в соответствии с разделом Error signaling технического описания DMA
int_global1	UART1	Сигнал UARTINTR
int_global2	UART2	Сигнал UARTINTR
int_global3	SSP1	Сигнал SSPINTR
int_global4	POWER	Сигнал прерывания от POWER Detecor
int_global5	WWDG	Сигнал прерывания от WWDG
int_global6	Timer1	Сигнал прерывания от Таймера TIM_STATUS и TIM_IE
int_global7	Timer2	Сигнал прерывания от Таймера

		TIM_STATUS и TIM_IE
int_global8	ADC	Сигналы прерываний от АЦП EOCIF_1 или AWOIF_1 или EOCIF_2 или AWOIF_2
int_global9	I2C	Сигнал I2C_INT
int_global10	BACKUP	Прерывание от ВКР и часов реального времени
int_global11	Внешнее прерывание 1	Сигнал EXT_INT0 Вывод PA[10] в основном режиме
int_global12	Внешнее прерывание 2	Сигнал EXT_INT1 Вывод PC[4], PB[6] в альтернативном режиме
int_global13	Внешнее прерывание 3	Сигнал EXT_INT2 Вывод PC[5], PB[7] в альтернативном режиме
int_global14	ADCIU	Прерывание от АЦП для измерения напряжений и токов
int_global15	ADCIU	Прерывание от АЦП для измерения напряжений и токов
int_global16	ADCIU	Прерывание от АЦП для измерения напряжений и токов
int_global17	L_BLOCK	Прерывание блока L-преобразования
int_global18	-	-
int_global19	SENSORS	Прерывание блока обработки датчиков безопасности
int_global20	CLK_MEASURE	Прерывание блока датчика частоты
int_global21	RANDOM	Прерывание блока генератора случайных чисел
int_global22	ISO7816	Прерывание блока UART ISO7816
int_global23	UART3	Сигнал UARTINTR
int_global24	SSP2	Сигнал SSPINTR
int_global25	SSP3	Сигнал SSPINTR
int_global26	Timer3	Сигнал прерывания от Таймера TIM_STATUS и TIM_IE
int_global27	Timer4	Сигнал прерывания от Таймера TIM_STATUS и TIM_IE
int_global28	UART4	Сигнал UARTINTR
int_global29	otp_corrupt	Сигнал прерывания от блока проверки контрольной суммы (CRC) OTP

Контроллер обработки внешних прерываний

В данном разделе приведено описание контроллера обработки внешних прерываний (Platform Level Interrupt Controller, PLIC).

Концептуальная структурная схема контроллера обработки внешних прерываний представлена ниже.

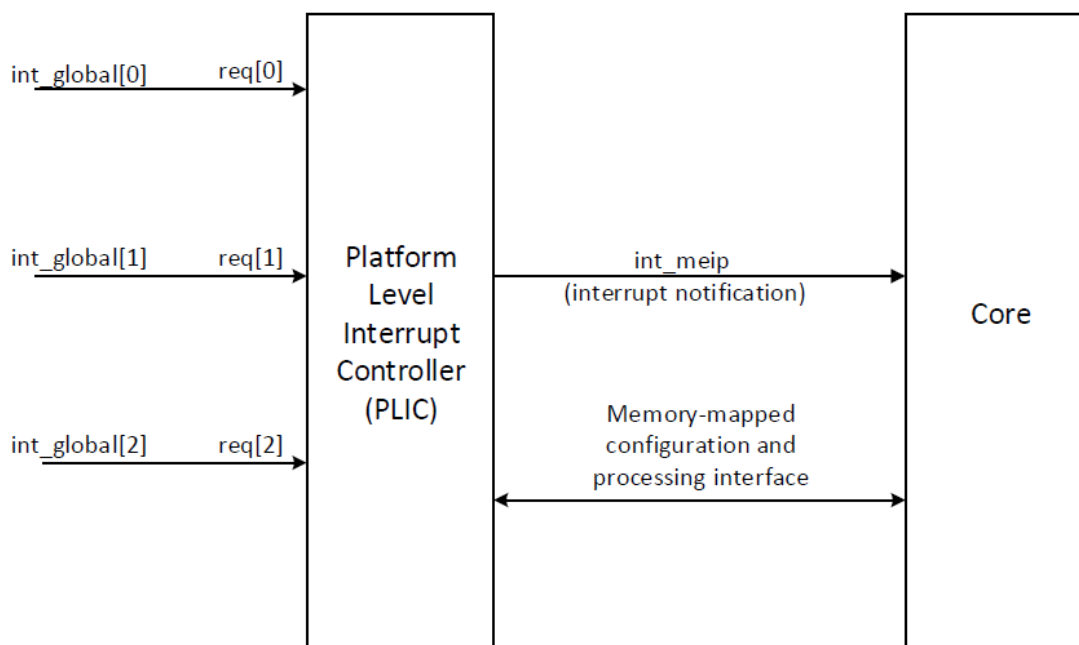


Рисунок 94. Структурная схема контроллера обработки внешних прерываний.

Внешние источники прерываний подключаются к входному интерфейсу процессорного комплекса через сигнал `int_global`. Сигналы, подключаемые к `int_global`, должны быть синхронны с тактовым сигналом ядра. Запросы прерываний `int_global` являются level-sensitive сигналами. `Int_global` подключается к `req` входу блока PLIC. Блок PLIC получает входные запросы прерываний и формирует запрос к ядру, сигнализирующий о наличии или отсутствии хотя бы одного прерывания, ожидающего обработки в ядре (сигнал `int_meip`).

Сигнал `int_meip` подключается в CSR регистр `mip` в бит MEIP (machine external interrupt pending, бит 11).

Сигнал `int_meip` используется только для передачи сообщения в ядро о наличии хотя бы одного прерывания в PLIC. Подтверждение факта «взятия» прерывания на исполнение (interrupt claiming) и оповещение о завершении выполнения прерывания (interrupt completion), а также включение/выключение посылки прерываний от отдельных источников выполняется через обращение к регистрам, отображенным в память (memory-mapped registers, MMR), командами чтения/записи памяти.

Детальная структурная схема PLIC представлена на рисунке ниже. Запросы прерываний передаются в PLIC через сигналы запроса прерывания (`req[i]`). Поступив в PLIC, каждый запрос попадает в блок проверки допустимости отправки прерывания, где проверяется разрешено ли данное прерывание для обработки (регистр IE, interrupt enable) и не обрабатывается ли уже данное прерывание (регистр IS, interrupt serving). Если прерывание разрешено (`IE=1`), не обрабатывается (`IS=0`) и его приоритет больше нуля, то оно становится

кандидатом на отправку для обработки. При наличии хотя бы одного кандидата, PLIC устанавливает в единицу сигнал `int_meip`, сообщая тем самым ядру о наличии прерывания, ожидающего обработки. Среди всех кандидатов в каждый момент времени выбирается одно прерывание с наибольшим приоритетом, идентификатор, которого считывается ядром, в момент чтения MMR-регистра ICC completion).

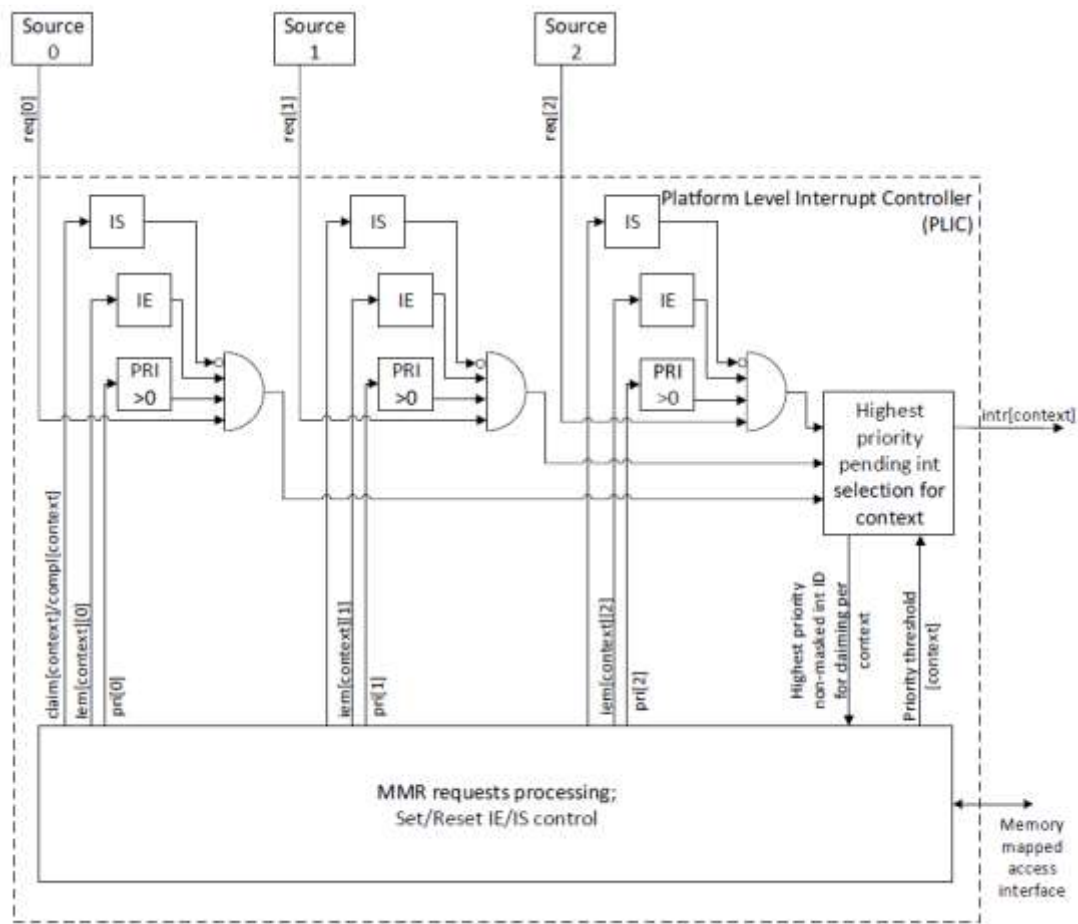


Рисунок 95. Структурная схема PLIC.

Основной интерфейс для взаимодействия с PLIC - это регистры, отображенные в память. Карта регистров PLIC представлена в таблице ниже.

Процессорный комплекс поддерживает 31-но внешнее прерывание. Прерывания в блоке PLIC нумеруются с 1-ого по 31-ое. Сигнал `int_global[0]` соответствует 1-му прерыванию, `int_global[30]` соответствует 31-му прерыванию. Нулевой номер зарезервирован и имеет семантику «нет прерывания».

Регистры PLIC

Адрес	Название	Доступ
0x0C00_0004	Регистр приоритета 1-ого прерывания (PRI1)	R/W
0x0C00_007C	Регистр приоритета 31-ого прерывания (PRI31)	R/W
0x0C00_1000	Регистр-маска наличия прерываний (IPM)	R
0x0C00_2000	Регистр-маска разрешения прерываний (IEM)	R/W
0x0C20_0000	Регистр порога приоритета прерываний (THR)	R/W
0x0C20_0004	Регистр взятия/завершения обработки прерывания (ICC)	R/W

Регистры PRI

Для каждого источника прерываний существует отдельный регистр PRI, позволяющий задать его приоритет. Процессорный комплекс поддерживает 7 уровней приоритета прерываний. Приоритеты прерываний нумеруются с 1 (низкий) по 7 (высокий). Значение приоритета 0 означает, что прерывание запрещено. После сброса значения приоритетов не определены, поэтому перед разрешением прерывания, необходимо обязательно сконфигурировать его приоритет.

Регистр приоритета 1-ого прерывания (PRI1)

Биты	Название	Доступ	Описание	Значение
3:0	pri1	R/W	Задаёт приоритет 1-ого прерывания (соответствует 0-му биту int_global). 0- прерывание запрещено, 1 –низкий приоритет, 7-высокий приоритет	x

Регистр приоритета 31-ого прерывания (PRI31)

Биты	Название	Доступ	Описание	Значение
3:0	pri31	R/W	Задаёт приоритет 31-ого прерывания (соответствует 30-му биту int_global). 0- прерывание запрещено, 1 –низкий приоритет, 7-высокий приоритет	x

Регистр IPM

Регистр представляет собой битовую маску. Единица в i-ом бите регистра указывает на наличие прерывания от (i-1) -го источника; ноль - на отсутствие прерывания. Если прерывание уже обрабатывается, бит соответствующего прерывания в маске сброшен. Запись в регистр запрещена. Попытка записи может вызывать исключение в ядре.

Регистр-маска наличия прерываний (IPM)

Биты	Название	Доступ	Описание	Значение
31:0	ipm	R	Единица в i-ом бите указывает на наличие прерывания от (i - 1) -го источника; ноль - на отсутствие прерывания. Если прерывание уже обрабатывается, бит соответствующего прерывания в маске равен 0. 0-ой бит регистра всегда равен нулю.	x

Регистры IEM

Регистр представляет собой битовую маску. Единица в i-ом бите указывает на разрешение отправки прерывания от (i - 1) -го источника в аппаратный поток исполнения; ноль - на запрет. После сброса PLIC значения всех битов в регистре IEM сброшены в 0. По готовности принимать прерывания, ПО должно разрешить обработку желаемых прерываний путем записи 1 в соответствующие биты регистра IEM. Рекомендуется устанавливать в 1 только те биты IEM, которые соответствуют действительно существующим источникам, подключенным к ядру, чтобы избежать появления прерываний от несуществующих источников.

Регистр маски разрешения прерываний (IEM)

Биты	Название	Доступ	Описание	Значение
------	----------	--------	----------	----------

31:0	iem	R/W	Единица в i-ом бите указывает на разрешение прерывания от (i – 1) -го источника; ноль - на запрет. 0-ой бит регистра всегда равен нулю.	0
------	-----	-----	---	---

Регистр THR

Контроллер прерываний маскирует (запрещает) все прерывания, значения приоритета которых, меньше либо равны порогу (значению регистра THR). Например, значение порога 0 разрешает все прерывания с ненулевым приоритетом, а значение порога 7 запрещает все прерывания.

Регистр порога приоритета прерываний (THR)

Биты	Название	Доступ	Описание	Значение
2:0	thr	R/W	Запрещает все прерывания, значения приоритета которых, меньше либо равны порогу. Например, значение порога 0 разрешает все прерывания с ненулевым приоритетом, а значение порога 7 запрещает все прерывания.	x

Регистр ICC

Регистр ICC (interrupt claiming/completion) отвечает за «взятие» прерывания на исполнение (claiming) и сообщение о завершении обработки прерывания (completion). Чтение регистра возвращает номер наиболее приоритетного прерывания, ожидающего обработки в аппаратном потоке исполнения, и переводит прерывание в статус обрабатываемого. При отсутствии прерываний, ожидающих обработки, чтение возвращает 0. Запись сигнализирует о завершении обработки прерывания, взятого ранее в обработку. Записываемое значение должно совпадать со считанным значением при взятии на обработку. Попытка записи другого значения оставит прерывание в статусе обрабатываемого.

Регистр взятия/завершения обработки прерывания (ICC)

Биты	Название	Доступ	Описание	Значение
31:0	icc	R/W	Чтение (claim) возвращает номер наиболее приоритетного прерывания, ожидающего обработки, переводит прерывание в состояние обрабатываемого. При отсутствии прерываний, ожидающих обработки, возвращает 0. Запись (complete) сигнализирует о завершении обработки прерывания, взятого ранее на обработку.	x

Программная модель обработки внешних прерываний

Концептуальная временная диаграмма обработки внешнего прерывания представлена на рисунке ниже. Источник прерываний (interrupt source) через PLIC сигнализирует в ядро о наличии прерывания. После выполнения необходимых аппаратных проверок о допустимости выполнения прерываний и выбора наиболее приоритетного прерывания (при готовности нескольких прерываний), PLIC формирует сигнал `int_meip` к ядру, указывающий на наличие прерывания, готового к обработке. Сигнал `int_meip` подключается в CSR регистр `mpir` в бит `MEIP` (machine external interrupt pending, бит 11).

Если прерывания не запрещены (в регистре `mie` установлен бит `MEIE` и в регистре `mstatus` установлен бит `MIE`), управление передается в обработчик прерывания (выполняется переход на адрес обработчика прерываний), а в регистр `mcause` записывается «причина», т.е. тип источника, вызвавшего прерывание. Для внешних прерываний тип устанавливается в значение `Machine external interrupt`.

Для «взятия» внешнего прерывания на исполнение (claiming) необходимо выполнить команду чтения по адресу MMR-регистра ICC. Эта транзакция сигнализирует в PLIC о том, что аппаратный поток принял прерывание к исполнению. После завершения обработки прерывания необходимо выполнить команду записи по адресу регистра ICC, что сигнализирует в PLIC о том, что обработка прерывания завершена.

Помимо использования модели обработки прерываний, описанной выше, для отслеживания наличия прерываний может быть использована модель опроса (poll model), в которой вместо «ожидания» появления прерывания и «заброса» исполнения программы в обработчик прерывания, ПО может самостоятельно проверять наличие запросов на прерывания путем чтения регистра `IPM`. ПО также может выполнять «взятие» прерывания на обработку путем чтения регистра `ICC`.

При использовании данной модели программирования надо учитывать, что чтение `IPM` не гарантирует, что последующее чтение `ICC` вернет прерывание с тем же номером. Последующее чтение `ICC` может вернуть прерывание с тем же номером или с другим номером, а также вернуть номер 0, указывающий на отсутствие прерывания.

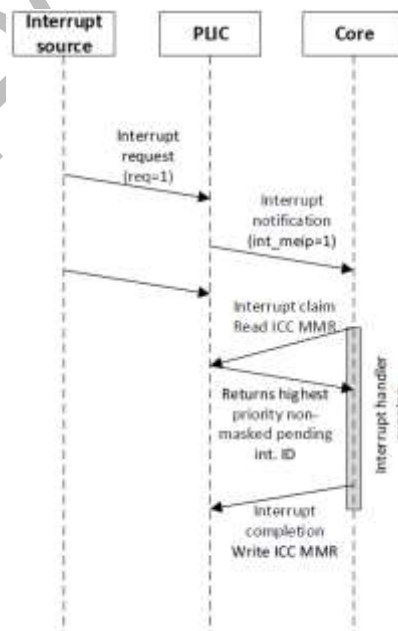


Рисунок 96. Временная диаграмма обработки прерывания

Подсистема отладки

Подсистема отладки ядра соответствует спецификации RISC-V External Debug Support Version 0.13. Поддерживаются следующие опции:

- Чтение/запись регистров общего назначения с помощью абстрактных команд.
- Read/Write программный буфер 64 байта.
- 4 регистра абстрактных данных.
- Память, доступная только по чтению (Debug ROM).

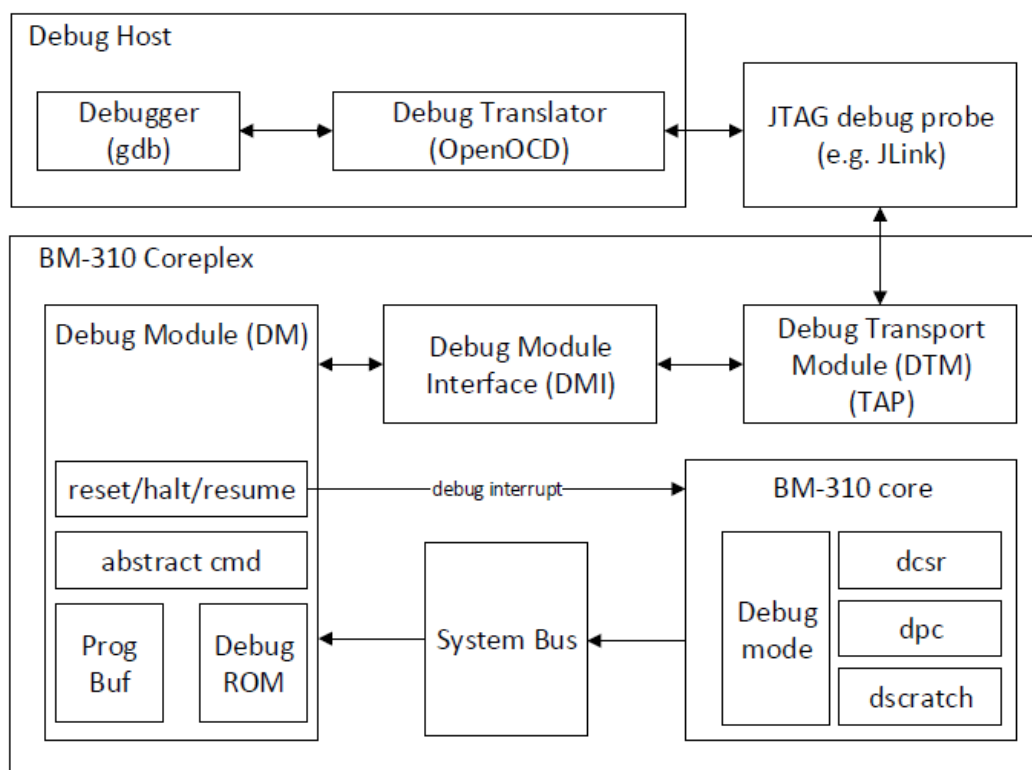


Рисунок 97. Архитектура системы отладки.

Debug Transport

Подсистема отладки состоит из блока DTM (Debug Transport Module) и блока DM (Debug Module). DTM связан с DM через блок DMI (Debug Module Interface). DTM содержит TAP (Test Access Port), соответствующий спецификации интерфейса JTAG. Размер IR - 5 бит. Доступ к DM через JTAG осуществляется через регистр DMI. Список доступных по JTAG регистров приведен в таблице ниже. Общая схема цепочки отладки процессорного ядра представлена на рисунке ниже. Переход из области синхросигнала JTAG в область синхросигнала ядра происходит в блоке DMI.

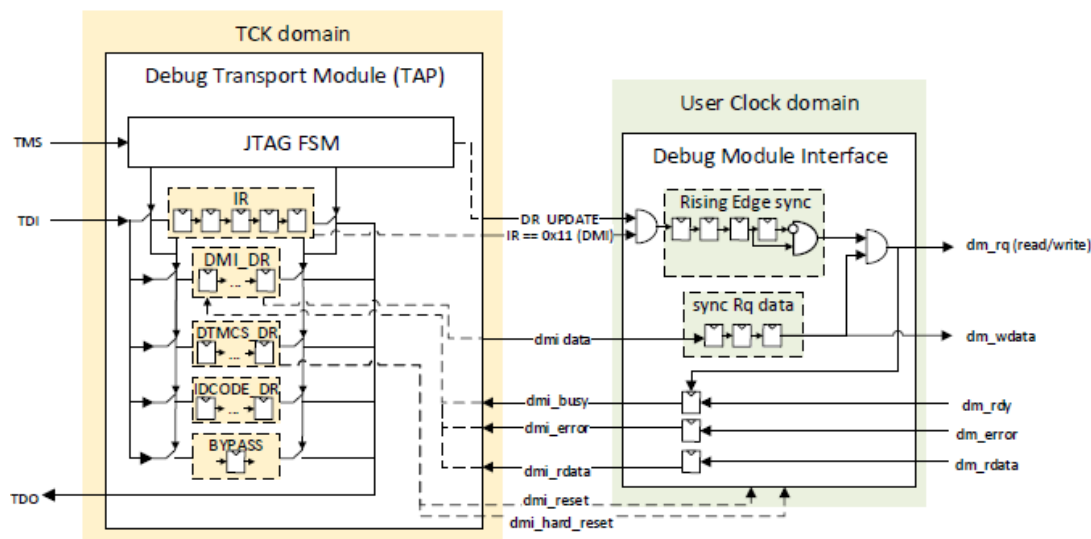


Рисунок 98. Интерфейс DM.

Регистры DMT

Адрес	Название	Доступ
0x00	BYPASS	R
0x01	Регистр идентификатора производителя микросхемы (IDCODE)	R
0x10	Регистр статуса и управления DTM (DTMCs)	R/W
0x11	Регистр доступа к DMI	R/W
0x12-0x1A	BYPASS	R/W

Регистр статуса и управления DTM (DTMCs)

Биты	Название	Доступ	Описание	Значение
17	dmihardreset	W1	Запись 1 сбрасывает DMI	
16	dmireset	W1	Запись 1 сбрасывает ошибку в DMI	
14:12	idle	R	Количество циклов, которое отладчик должен провести в состоянии IDLE, между запросом и чтением результата из DMI, для того, чтобы избежать состояния busy	5
11:10	dmistat	R	Состояние DMI: 0: Ошибок нет 1: См.2 2: Ошибка выполнения операции 3: Операция ещё не выполнена (состояние busy)	
9:4	abits	R	Размер поля адреса регистра DMI	7
3:0	version	R	Версия реализованной спецификации 0: 0.11 1: 0.13	1

Регистр доступа к DMI

Биты	Название	Доступ	Описание	Значение
abits+33 :34	address	R/W	Адрес запроса на DMI	
33:2	data	R/W	Данные для записи, отправляемые в DM во время состояния Update-DR, либо данные, считанные из DM	
1:0	op	R/W	Код операции. При записи: 0: Пустой запрос. В DM ничего не посылается 1: Чтение по адресу 2: Запись по адресу При чтении: 0: Предыдущий запрос завершен успешно 2: Предыдущий запрос вернул ошибку 3: Предыдущий запрос ещё не выполнен. Чтобы сбросить это состояние необходим сброс DMI.	

Взаимодействие отладчика с DM происходит через DMI путем чтения и записи регистров DM. Для этого используется регистр DMI в DTM. Значение поля abits равно 7. Список доступных по DMI регистров приведен в таблице ниже.

Регистры отладчика DM

Адрес	Название	Доступ
0x04-0x07	Abstract Data 0-3	R/W
0x10	DM control	R/W
0x11	DM status	R/W
0x12	Hart info	R/W
0x16	Abstract Control and Status	R/W
0x17	Abstract Command	R/W
0x18	Abstract Command Autoexec	R/W
0x20-0x2F	Program Buffer 0-15	R/W

Debug Mode

Взаимодействие процессорного ядра с DM происходит с помощью чтения/записи отображенных в памяти регистров (DM MMRs (Memory Mapped Registers)). Карта памяти DM представлена в таблице ниже.

Карта памяти DM

Адрес	Название	Доступ
0x0000_0000	Safe zero	R/W
0x0000_0100	Halted ack	W
0x0000_0104	Going ack	W
0x0000_0108	Resume ack	W

0x0000_010C	Exception	W
0x0000_0300	Where to	R
0x0000_0338	Abstract command 0	R
0x0000_033C	Abstract command 1	R
0x0000_0340-0x37C	Program buf	R/W
0x0000_0380-0x38C	Abstract data	R/W
0x0000_0400	CPU flags	R
0x0000_0800-0xFFFF	Debug ROM	R

CPU flags

Биты	Название	Доступ	Описание	Значение
0	Going	R	Запрос на выполнение абстрактной команды. Находясь в режиме отладки, ядро считывает данное поле и делает переход по адресу первой абстрактной команды (0x338)	
1	Resumereq	R	Запрос на выход из режима отладки. Находясь в режиме отладки, ядро считывает данное поле и выходит из режима отладки, если его значение равно 1. Исполнение программы продолжается со значения PC, сохраненного в регистре dpc	

Для взаимодействия с внешним отладчиком внутри процессорного ядра реализован специальный режим исполнения: режим отладки. В таблице ниже представлен список контрольных регистров режима отладки. Процессор имеет право читать и писать контрольные регистры отладки только в режиме отладки. В любом другом режиме попытка доступа к данным регистрам вызовет исключение. Исполнение в режиме отладки имеет следующие свойства:

1. Операции обращения к контрольным регистрам и к памяти происходят также как в режиме привилегированности M.
2. Любые прерывания не обрабатываются.
3. Исключения не обновляют значения контрольных регистров.
4. Все инструкции смены уровня привилегированности не обрабатываются. Исключение - инструкция *ebreak*. При исполнении инструкции *ebreak* в режиме отладки процессор выполняет переход на начало Debug ROM, без обновления dpc и dcsr.

Контрольные регистры режима отладки

Адрес	Название	Доступ
0x7B0	Регистр управления режимом отладки dcsr	R/W
0x7B1	Регистр счетчика команд dpc	R/W
0x7B2	Дополнительный регистр режима отладки dscratch	R/W

Регистр управления режимом отладки dcsr

Биты	Название	Доступ	Описание	Значение
31:28	version	R/W	Версия: 0: Режим отладки не поддерживается 4: Режим отладки поддерживается и соответствует спецификации версии 0.13	4
15	ebreakm	R/W	Значение 1: При исполнении инструкции <i>ebreak</i> в режиме M процессор переходит в режим отладки.	0
12	ebreaku	R/W	Значение 1: При исполнении инструкции <i>ebreak</i> в режиме U процессор переходит в режим отладки.	0
8:6	cause	R	Причина перехода в режим отладки: 1: Выполнена инструкция <i>ebreak</i> 3: Прерывание от DM 4: Исполнена очередная инструкция в режиме пошагового исполнения	0
2	step	R/W	Значение 1: режим пошагового исполнения включен. В данном режиме после выхода из режима отладки процессор исполняет очередную инструкцию пользовательского приложения и снова переключается в режим отладки. При пошаговом исполнении прерывания выключены	0
1:0	prv	R/W	Режим привилегированности, в котором находилось ядро перед переключением в режим отладки. Отладчик может изменить значение этого поля, чтобы изменить режим процессора после выхода из режима отладки	0

Регистр счетчика команд dpc

Биты	Название	Доступ	Описание	Значение
31:0	dpc	R/W	Сохраненное значение счетчика команд. При входе в режим отладки в данный регистр сохраняется текущее значение счетчика команд. Запись в данный регистр возможна только в	

			режиме отладки. При выходе из режима отладки исполнение продолжается с адреса, сохраненного в данном регистре	
--	--	--	---	--

Дополнительный регистр режима отладки dscratch

Биты	Название	Доступ	Описание	Значение
31:0	dscratch	R/W	Дополнительный регистр, используемый DM для сохранения промежуточной информации	

После входа в режим отладки процессорное ядро выполняет переход по адресу начала Debug ROM (адрес 0x800). Программный код, сохраненный в Debug ROM, осуществляет опрос флагов DM. При определении запроса на исполнение абстрактной команды (флаг “Going” в регистре 0x400) происходит переход исполнения на адрес первой абстрактной команды (регистр по адресу 0x338). В случае возникновения исключения во время исполнения абстрактной команды или кода из программного буфера ядро переходит по адресу внутри Debug ROM, где сохранен код обработчика исключений. В данном обработчике находится инструкция записи по адресу 0x10c. После чего отладчик может считать значение регистра Abstract Control and Status и определить факт возникновения исключения в процессе выполнения абстрактной команды. Если при опросе считывается запрос на продолжение исполнения пользовательского приложения (флаг “Resumereq”), то выполняется код, осуществляющий переход по адресу, сохраненному в регистре dpc.

Любые операции отладчика: считывание/запись регистров, считывание/запись памяти, установка точек остановки в коде, последовательное исполнение - осуществляются с помощью модификации кода в программном буфере и исполнения абстрактных команд.

Сторожевые таймеры

Описание регистров блока сторожевых таймеров

Таблица 449 Описание регистров блока сторожевых таймеров

Базовый Адрес	Название	Описание
0x4005_0000	IWDG	Сторожевой таймер IWDG
Смещение		
0x00	IWDG_KR[15:0]	Регистр Ключа
0x04	IWDG_PR[2:0]	Делитель частоты сторожевого таймера
0x08	IWDG_PRL[11:0]	Регистр основания счета сторожевого таймера
0x0C	IWDG_SR[1:0]	Регистр статуса сторожевого таймера

Таблица 450 Описание регистров блока сторожевых таймеров

Базовый Адрес	Название	Описание
0x4004_8000	WWDG	Сторожевой таймер WWDG
Смещение		
0x00	WWDG_CR[7:0]	Регистр управления
0x04	WWDG_CFR[9:0]	Регистр конфигурации
0x08	WWDG_SR[0]	Регистр статуса

IWDG – это независимый сторожевой таймер, который считает сверху вниз до нуля, начиная со значения, которое мы в нём установим. Как только счётчик дойдёт до нуля, то от сторожевого таймера поступит сигнал сброса, который заставит систему перезагрузиться. Но чтобы этого не произошло, мы должны заблаговременно перезагрузить IWDG или восстановить регистр, внося заново в него требуемое число и тогда таймер начнёт снова обратный отсчёт.

Рассмотрим его блок-схему.

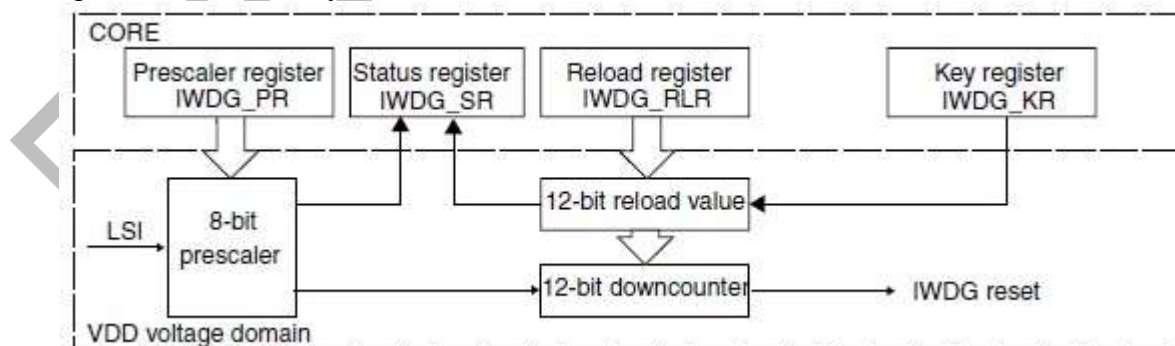


Рисунок 99. Блок-схема независимого сторожевого таймера.

Из блок-схемы мы видим, что тактируется таймер от отдельного низкоскоростного источника тактовых сигналов LSI (low-speed clock), от которого тактовые сигналы поступают в 8-битный делитель, коэффициент которого устанавливается с помощью

регистра IWDG_PR. В регистр IWDG_PLR мы заносим значение, от которого таймер при старте начнёт обратный отсчёт. Также существует регистр статуса IWDG_SR, в котором в момент изменения значения в регистре предделителя IWDG_PR, а также в момент занесения или обновления значения в регистре IWDG_PLR устанавливается соответствующий бит. Предварительно в ключевой регистр IWDG_KR для доступа к данным операциям изменений мы заносим определённое ключевое число. После того, как 12-битный обратный счётчик досчитает до 0, IWDG выдаст на своём выходе сигнал на перезагрузку системы.

Чтобы перезагрузка системы не произошла, мы должны своевременно подать команду на перезагрузку счётчика, чтобы он заново начал обратный отсчёт. Тем самым и обеспечивается контроль над зависанием программы в определённых точках, где такое возможно. Если мы не перезагрузим IWDG вовремя, это будет означать, что программа зависла и система перезагрузится.

IWDG_KR

Таблица 451 Регистр IWDG_KR

Номер	31...16	15...0
Доступ		W
Сброс		0
	-	KEY[15:0]

Таблица 452 Описание бит регистра IWDG_KR

№	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...16		Зарезервировано
15...0	KEY[15:0]	Значение ключа (только запись, читается 0000h). Эти биты должны перезаписываться программно через определённые интервалы ключевым значением AAAAh, в противном случае сторожевой таймер генерирует сброс, если таймер достиг значения нуля. Запись ключевого значения 5555h разрешает доступ по записи к регистрам IWDG_PR и IWDG_RLR. Запись ключевого значения CCCCCh разрешает работу сторожевого таймера (за исключением, если сторожевой таймер разрешается аппаратно битами конфигурации).

IWDG_PR

Таблица 453 Регистр IWDG_PR

Номер	31...3	2	1	0
Доступ	U	R/W	R/W	R/W
Сброс		0	0	0
	-	PR2	PR1	PR0

Таблица 454 Описание бит регистра IWDG_PR

№	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...3		Зарезервировано
2...0	PR[2:0]	Делитель частоты сторожевого таймера. 000 – делитель на 4 001 – делитель на 8 010 – делитель на 16 011 – делитель на 32 100 – делитель на 64 101 – делитель на 128 110 – делитель на 256 111 – делитель на 256 Чтение и запись этого регистра правомерна только, если бит PVU=0 в регистре IWDG_SR.

IWDG_RLR

Таблица 455 Регистр IWDG_RLR

Номер	31...12	11...0
Доступ	U	R/W
Сброс		1
	-	RLR[11:0]

Таблица 456 Описание бит регистра IWDG_RLR

№	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...12		Зарезервировано
11...0	RLR[11:0]	Значение перезагрузки сторожевого таймера. Значение этих бит по доступу защищено с помощью регистра IWDG_KR. Эти биты записываются программно и определяют значение, загружаемое в сторожевой таймер в момент записи значения AAAAh в регистр IWDG_KR. Сторожевой таймер декрементируется, начиная с этого значения. Период таймаута сторожевого таймера - функция от этого значения и делителя частоты. Чтение и запись этого регистра правомерна только, если бит RVU=0 в регистре IWDG_SR.

IWDG_SR

Таблица 457 Регистр IWDG_SR

Номер	31...2	1	0
Доступ	U	R	R
Сброс		0	0
	-	RVU	PVU

Таблица 458 Описание бит регистра IWDG_SR

№	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...2		Зарезервировано
1	RVU	Флаг обновления значения сторожевого таймера. Этот бит устанавливается аппаратно и служит признаком того, что обновляется значение сторожевого таймера из регистра перезагрузки. Этот бит сбрасывается, если обновление завершено. Значение регистра перезагрузки может быть обновлено только, если этот бит равен нулю.
0	PVU	Флаг обновления делителя частоты сторожевого таймера. Этот бит устанавливается аппаратно и служит признаком того, что обновляется значение делителя частоты сторожевого таймера. Этот бит сбрасывается, если обновление завершено. Значение регистра делителя частоты может быть обновлено только, если этот бит равен нулю.

WWDG в данном контроллере – это оконный сторожевой таймер, счётчик которого считает сверху вниз, начиная со значения, которое мы вносим в регистр WWDG_CR (биты T[6:0]), но не до нуля, а до 63 (0x3F). Только кроме значения, с которого считает WWDG, и значения 0x63, до которого считает данный сторожевой таймер, существует ещё одно промежуточное значение, находящееся между данными величинами. Оно также является задаваемой величиной и, таким образом, образуется окно между данной величиной и значением 0x3F, поэтому данный сторожевой таймер и является оконным. Подобно IWDG, сторожевой таймер WWDG также, досчитав до 0x3F, даст команду на перезагрузку системы, если же мы, конечно, не перезагрузим данный оконный сторожевой таймер своевременно. Только перезагрузить мы должны его не только не позже, чем его счётчик досчитает вниз до 0x3F, но ещё и не раньше, чем он досчитает до промежуточной величины, которая также заносится в регистр WWDG_CFR (биты W[6:0]). Данная величина поэтому должна находиться в интервале от числа 0x3F до заданного значения счётчика, с которого он начинает обратный отсчёт. Таким образом, мы получаем возможность проверки определённых временных интервалов хода нашей программы.

Рассмотрим блок-схему оконного сторожевого таймера.

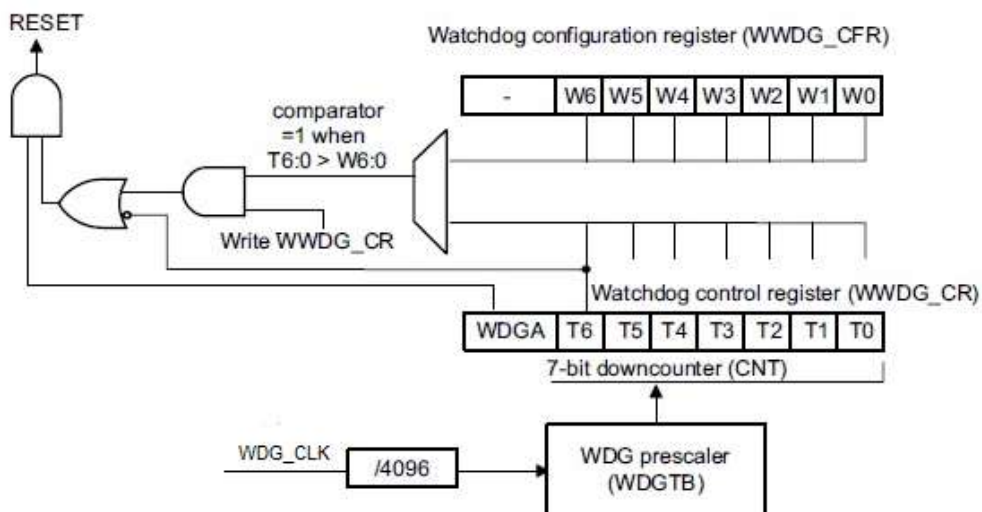


Рисунок 100. Блок-схема оконного сторожевого таймера.

Здесь мы можем уже наглядно наблюдать, как происходит работа WWDG. Тактовые сигналы счётчика идут через первый делитель с постоянным коэффициентом деления 4096 с шины WDG_CLK на второй, регулируемый делитель, значение коэффициента деления которого мы можем устанавливать с помощью битов регистра WGTB. После второго делителя тактовые сигналы попадают на счётчик обратного отсчёта.

Число, которое находится в битах T6:T0 регистра WWDG_CR после старта оконного таймера начинает инкрементироваться, как только оно достигнет значения 0x3F, то есть как только бит T6 очистится, на один из мультиплексоров пойдёт сигнал о перезагрузке системы. Но на то он и мультиплексор, что сигнал о перезагрузке он может получить не только от очистки бита T6. Аналогичный сигнал он может получить от компаратора, в котором сравниваются значения битов T6:T0 регистра WWDG_CR со значениями битов W6:W0 регистра WWDG_CFR. И, если первое число больше второго, то есть если счётчик обратного отсчёта ещё не досчитал до значения первой границы, а в это время мы пытались перезагрузить оконный сторожевой таймер, то мультиплексор и получит данный сигнал, который также заставит систему перезагрузиться.

Также процесс работы оконного сторожевого таймера показан и на графике.

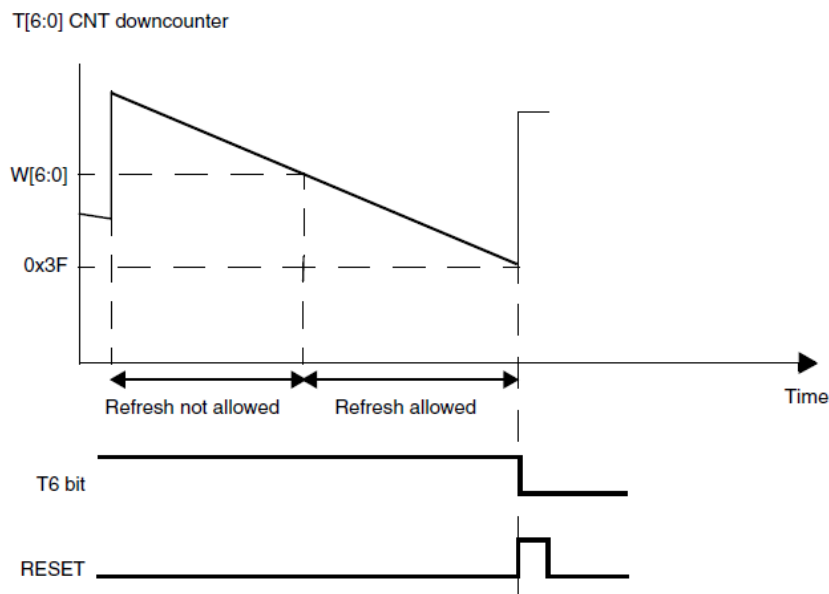


Рисунок 101. Временная диаграмма оконного сторожевого таймера. Можно рассчитать время работы WWDG от старта до окончания счёта и перезагрузки по следующей формуле:

$$t_{\text{WWDG}} = t_{\text{WDG_CLK}} \times 4096 \times 2^{\text{WGTB}} \times (T[5:0] + 1), \text{ мс}$$

t_{WWDG} – таймаут

$t_{\text{WDG_CLK}}$ – период синхросигнала APB шины в мс.

WWDG_CR

Таблица 459 Регистр WWDG_CR

Номер	31...8	7	6...0
Доступ	U	R/S	R/W
Сброс		0	1
	-	WDGA	T6...T0

Таблица 460 Описание бит регистра WWDG_CR

№	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...18		Зарезервировано
7	WDGA	Бит активации. Этот бит устанавливается программно и очищается только аппаратно при сбросе. Когда WDGA=1, сторожевой таймер может генерировать сброс. 0 – сторожевой таймер отключен 1 - сторожевой таймер включен
6..0	T[6:0]	Значение семиразрядного счётчика (от старших разрядов к младшим). Эти биты содержат значение сторожевого таймера, который декрементируется каждые $4096 \times 2^{\text{WGTB}}$ циклов частоты WDG_CLK периферийной шины APB

WWDG_CFR

Таблица 461 Регистр WWDG_CFR

Номер	31...10	9	8	7	6...0
Доступ	U	R/S	R/W	R/W	R/W
Сброс		0	0	0	1
	-	EWI	WDGTB1	WDGTB0	W6...W0

Таблица 462 Описание бит регистра WWDG_CFR

№	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...10		Зарезервировано
9	EWI	Раннее предупреждающее прерывание. Если бит установлен, то разрешается генерация прерывания при достижении сторожевым таймером значения 40h. Прерывание запрещается только аппаратным сбросом.
8..7	WGTB[1:0]	Делитель частоты сторожевого таймера. 00 – частота таймера (WDG_CLK / 4096) /1 01 – частота таймера (WDG_CLK / 4096) /2 10 – частота таймера (WDG_CLK / 4096) /4 11 – частота таймера (WDG_CLK / 4096) /8
6..0	W[6:0]	Значение окна. Эти биты содержат значение окна, в пределах которого возможна инициализация битов T[6:0] значением в пределах 40h-7Fh. Если происходит инициализация битов в момент T>W, то формируется сброс на выходе RESET. Если таймер достигнет значения T=3Fh, то также формируется сброс.

WWDG_SR

Таблица 463 Регистр WWDG_SR

Номер	31...2	1	0
Доступ	U	R	R/C
Сброс		1	0
	-	WDG_WEC	EWIF

Таблица 464 Описание бит регистра WWDG_SR

№	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...2		Зарезервировано
1	WDG_WEC	Флаг окончания записи в регистры WWDG_CR и WWDG_CFR. 1-запись завершена 0-запись не завершена
0	EWIF	Флаг раннего предупреждающего прерывания. Этот бит устанавливается аппаратно, когда сторожевой таймер достигает значения 40h. Бит очищается

		программно записью нуля. Запись единицы не влияет. Этот бит также устанавливается, если прерывание запрещено $EWI=0$.
--	--	--

ПРЕДВАРИТЕЛЬНО

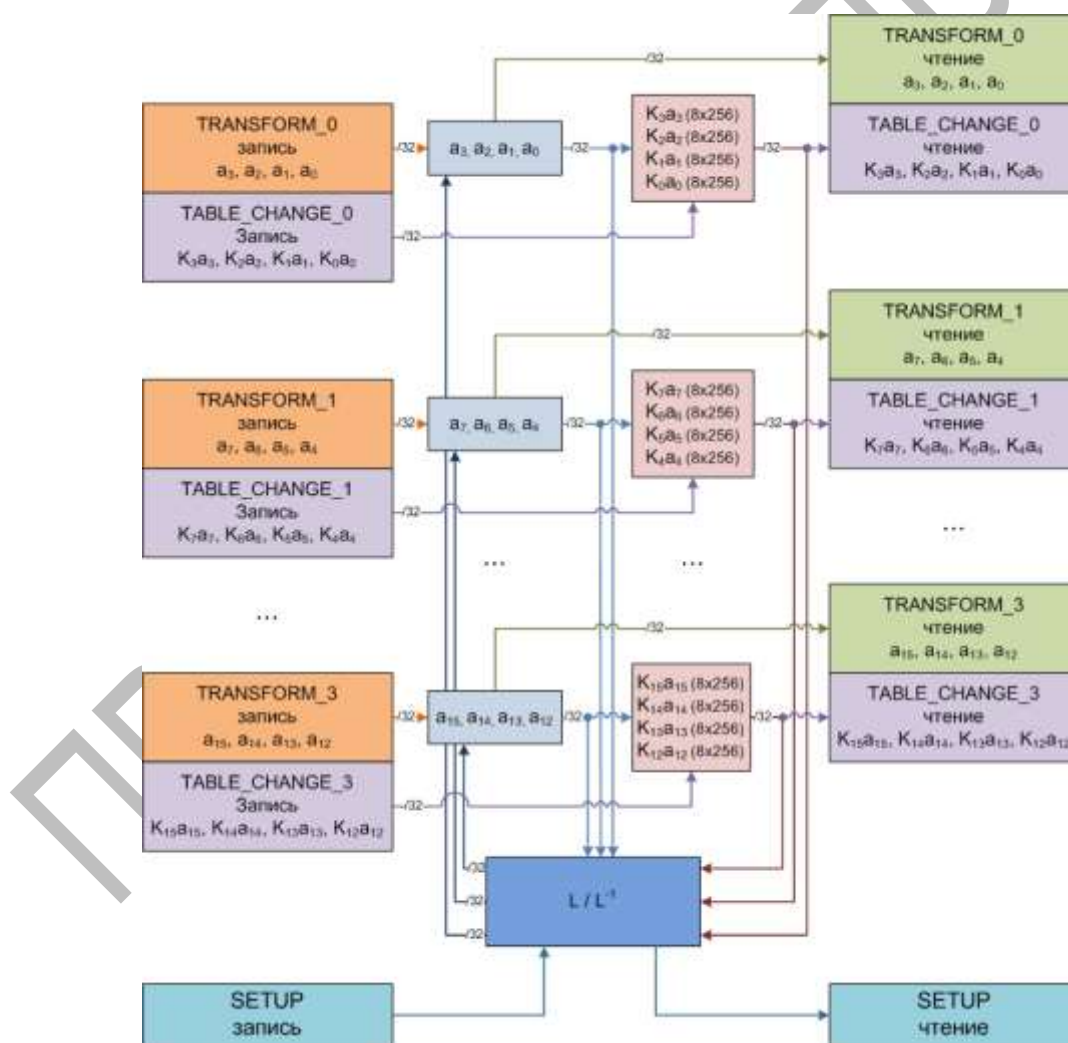
Блок прямого и обратного L-преобразования

Блок преобразования L / L^{-1} соответствует ГОСТ Р34.12-2015.

Особенности

- Возможность задания направления преобразования L / L^{-1}
- Возможность выбора числа R преобразований (0 - 31)
- Возможность задания коэффициентов l преобразования через 16 таблиц соответствия $a \rightarrow K*a$
- Время выполнения преобразований равно числу преобразований R плюс 1 такт
- АНВ подключение
- Сигнал занятости модуля текущим преобразованием

Рисунок 102. Структурная схема L-блока.



Регистры модуля

Базовый Адрес	Название	Описание
0x6000_4000	L_BLOCK	Блок прямого и обратного L преобразования
Смещение		
0x00	TRANSFORM_0	Регистр преобразования 1
0x04	TRANSFORM_1	Регистр преобразования 2
0x08	TRANSFORM_2	Регистр преобразования 3
0x0C	TRANSFORM_3	Регистр преобразования 4
0x10	TABLE_CHANGE_0	Регистр изменения таблицы 1
0x14	TABLE_CHANGE_1	Регистр изменения таблицы 2
0x18	TABLE_CHANGE_2	Регистр изменения таблицы 3
0x1C	TABLE_CHANGE_3	Регистр изменения таблицы 4
0x20	SETUP	Регистр настройки преобразования
0x24	M_SCTRL	

Регистр преобразования n – ого слова данных $TRANSFORM_n$ (адрес: $0x00 + 4 * n \dots 0x0c$)

Бит	Сокращенное наименование	Функция	Доступ
31:0	DATA	Запись в данный регистр передает n - ое слово для L / L^{-1} преобразования (запуск преобразования выполняется при записи в старший байт последнего регистра, результат преобразования доступен через $(SET_R_COUNT + 1)$ тактов (см. <i>регистр настройки преобразования</i>)). Запись в данный регистр выбирает ячейки для изменения в n - ой группы таблиц коэффициентов l преобразования (см. <i>регистры изменения таблицы</i>). Чтение из данного регистра возвращает n - ое слово результата L / L^{-1} преобразования.	RW

Регистр изменения таблицы $TABLE_CHANGE_n$ (адрес: $0x10 + 4 * n \dots 0x1c$)

Бит	Сокращенное наименование	Функция	Доступ
7:0	TABLE_VAL_n0	Запись в данное поле изменяет значение перевода $a \rightarrow K*a$, 0 - го (младшего) байта n - го входного слова (значение a выбирается записью значение в $TRANSFORM_n$). Изменения необходимо производить в режиме прямого преобразования с нулевым количеством преобразований R (см. <i>регистр настройки преобразования</i>). Чтение данного поля возвращает текущее значение перевода $a \rightarrow K*a$, 0 - го (младшего) байта n - го входного слова (значение a выбирается записью значение в $TRANSFORM_n$).	RW

Бит	Сокращенное наименование	Функция	Доступ
15:8	TABLE_VAL_n1	Запись в данное поле изменяет значение перевода $a \rightarrow K*a, 1$ - го байта n - го входного слова (значение a выбирается записью значение в TRANSFORM_n). Изменения необходимо производить в режиме прямого преобразования с нулевым количеством преобразований R (см. <i>регистр настройки преобразования</i>). Чтение данного поля возвращает текущее значение перевода $a \rightarrow K*a, 1$ - го байта n - го входного слова (значение a выбирается записью значение в TRANSFORM_n).	RW
23:16	TABLE_VAL_n2	Запись в данное поле изменяет значение перевода $a \rightarrow K*a, 2$ - го байта n - го входного слова (значение a выбирается записью значение в TRANSFORM_n). Изменения необходимо производить в режиме прямого преобразования с нулевым количеством преобразований R (см. <i>регистр настройки преобразования</i>). Чтение данного поля возвращает текущее значение перевода $a \rightarrow K*a, 2$ - го байта n - го входного слова (значение a выбирается записью значение в TRANSFORM_n).	RW
31:24	TABLE_VAL_n3	Запись в данное поле изменяет значение перевода $a \rightarrow K*a, 3$ - го (старшего) байта n - го входного слова (значение a выбирается записью значение в TRANSFORM_n). Изменения необходимо производить в режиме прямого преобразования с нулевым количеством преобразований R (см. <i>регистр настройки преобразования</i>). Чтение данного поля возвращает текущее значение перевода $a \rightarrow K*a, 3$ - го (старшего) байта n - го входного слова (значение a выбирается записью значение в TRANSFORM_n).	RW

Регистр настройки преобразования SETUP (адрес: 0x20)

Бит	Сокращенное наименование	Функция	Доступ
4:0	SET_R_COUNT	Настройка количества R / R^{-1} преобразований при реализации L / L^{-1} преобразования. Нулевое значение - выключение преобразования. Число R / R^{-1} преобразований выбирается на все L / L^{-1} преобразование при старте (при записи в старший байт регистра TRANSFORM_3). Изменение данного поля не влияет на результат выполняемого преобразования.	RW
7:5	-	Зарезервировано	RO
12:8	CURR_R_COUNT	Число оставшихся R / R^{-1} в текущем, выполняем L / L^{-1} преобразовании. Нулевое значение означает что преобразование не начато или уже выполнено. Запись в данное поле игнорируется.	RO

15:13	-	Зарезервировано	RO
16	SET_L_DIR	Направление преобразования. 0 - прямое, L преобразование 1 - обратное, L⁻¹ преобразование Направление выбирается на все преобразование при старте (при записи в старший байт регистра TRANSFORM_3). Изменение данного поля не влияет на результат выполняемого преобразования.	RW
17	CURR_L_DIR	Направление текущего, выполняемого преобразования. Запись а данное поле игнорируется	RO
31:18	-	Зарезервировано	RO

Инициализация

- В регистр SETUP задать значение 0x00000000
- В регистры TRANSFORM_0 ... TRANSFORM_3 задать значение 0x00_00_00_00
- В регистры TABLE_CHANGE_0 задать значение 0xTT_PP_RR_SS, где TT = K_{3a}, PP = K_{2a}, RR = K_{1a}, SS = K_{0a}, для a = 0x00
- В регистры TABLE_CHANGE_1 задать значение 0xTT_PP_RR_SS, где TT = K_{7a}, PP = K_{6a}, RR = K_{5a}, SS = K_{4a}, для a = 0x00
- В регистры TABLE_CHANGE_2 задать значение 0xTT_PP_RR_SS, где TT = K_{11a}, PP = K_{10a}, RR = K_{9a}, SS = K_{8a}, для a = 0x00
- В регистры TABLE_CHANGE_3 задать значение 0xTT_PP_RR_SS, где TT = K_{15a}, PP = K_{14a}, RR = K_{13a}, SS = K_{12a}, для a = 0x00
- В регистры TRANSFORM_0 ... TRANSFORM_3 задать значение 0x01_01_01_01
- В регистры TABLE_CHANGE_0 задать значение 0xTT_PP_RR_SS, где TT = K_{3a}, PP = K_{2a}, RR = K_{1a}, SS = K_{0a}, для a = 0x01
- В регистры TABLE_CHANGE_1 задать значение 0xTT_PP_RR_SS, где TT = K_{7a}, PP = K_{6a}, RR = K_{5a}, SS = K_{4a}, для a = 0x01
- В регистры TABLE_CHANGE_2 задать значение 0xTT_PP_RR_SS, где TT = K_{11a}, PP = K_{10a}, RR = K_{9a}, SS = K_{8a}, для a = 0x01
- В регистры TABLE_CHANGE_3 задать значение 0xTT_PP_RR_SS, где TT = K_{15a}, PP = K_{14a}, RR = K_{13a}, SS = K_{12a}, для a = 0x01
- ...
- В регистры TRANSFORM_0 ... TRANSFORM_3 задать значение 0xFF_FF_FF_FF
- В регистры TABLE_CHANGE_0 задать значение 0xTT_PP_RR_SS, где TT = K_{3a}, PP = K_{2a}, RR = K_{1a}, SS = K_{0a}, для a = 0xFF
- В регистры TABLE_CHANGE_1 задать значение 0xTT_PP_RR_SS, где TT = K_{7a}, PP = K_{6a}, RR = K_{5a}, SS = K_{4a}, для a = 0xFF
- В регистры TABLE_CHANGE_2 задать значение 0xTT_PP_RR_SS, где TT = K_{11a}, PP = K_{10a}, RR = K_{9a}, SS = K_{8a}, для a = 0xFF
- В регистры TABLE_CHANGE_3 задать значение 0xTT_PP_RR_SS, где TT = K_{15a}, PP = K_{14a}, RR = K_{13a}, SS = K_{12a}, для a = 0xFF
- В регистр SETUP задать значение необходимого преобразования, направление и число повторений преобразования R / R⁻¹

Выполнение преобразования

В регистры TRANSFORM_0 ... TRANSFORM_3 записать данные для преобразования. При записи в старший байт регистра TRANSFORM_3 автоматически запускается преобразование. Результат преобразования доступен через (SET_R_COUNT + 1) тактов, SET_R_COUNT - значение поля регистра SETUP. Так же ход преобразование можно отслеживать по счетчику CURR_R_COUNT, регистра SETUP. При достижении счетчиком значения 0, преобразование закончено. Результат преобразования считывается из регистров TRANSFORM_0 ... TRANSFORM_3. Для смены направления преобразования перед записью регистров TRANSFORM_0 ... TRANSFORM_3 необходимо изменить значение регистра SETUP. Изменять значение регистра SETUP можно сразу после старта очередного преобразования, новое значение будет применено к следующему преобразованию.

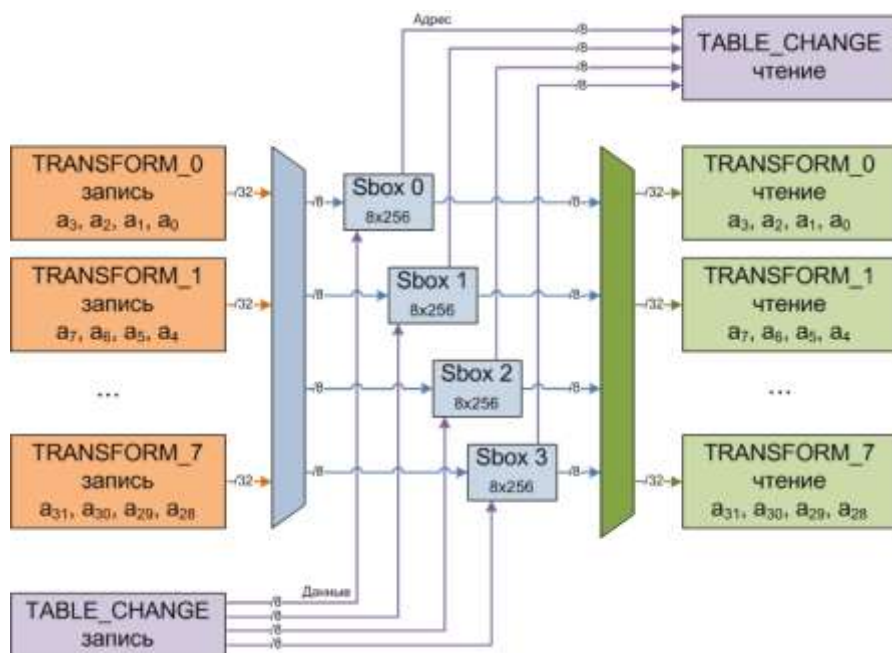
Блок замены S-box

Выполняет ускорение типовой задачи блочной замены. При работе данного блока производится преобразование 8bit → 8bit по таблице из 256 элементов. Каждое входное число заменяется на число из ячейки таблицы с индексом соответствующим входному числу.

Особенности

- Возможность задания таблицы замены 8bit → 8bit
- Обработка 32 битного слова за один такт
- Прием до восьми 32 битных слов с конвейерной обработкой и сохранением результата преобразования
- АНВ подключение

Рисунок 103. Структурная схема S-блока.



Регистры модуля

Базовый Адрес	Название	Описание
0x6000_8000	S_BLOCK0	Блок замены S-box0
0x6000_C000	S_BLOCK1	Блок замены S-box1
0x6001_0000	S_BLOCK2	Блок замены S-box2
0x6001_4000	S_BLOCK3	Блок замены S-box3
0x6001_8000	S_BLOCK4	Блок замены S-box4
0x6001_C000	S_BLOCK5	Блок замены S-box5
0x6002_0000	S_BLOCK6	Блок замены S-box6
0x6002_4000	S_BLOCK7	Блок замены S-box7
Смещение		
0x00	TRANSFORM_0	Регистр преобразования 1
0x04	TRANSFORM_1	Регистр преобразования 2
0x08	TRANSFORM_2	Регистр преобразования 3
0x0C	TRANSFORM_3	Регистр преобразования 4
0x10	TRANSFORM_4	Регистр преобразования 5
0x14	TRANSFORM_5	Регистр преобразования 6
0x18	TRANSFORM_6	Регистр преобразования 7
0x1C	TRANSFORM_7	Регистр преобразования 8
0x20	TABLE_CHANGE	Регистр изменения таблицы
0x24	M_SCTRL	

Регистр преобразования n – о го слова данных $TRANSFORM_n$ (адрес: $0x00 + 4 * n \dots 0x1C$)

Бит	Сокращенное наименование	Функция	Доступ
31:0	DATA	Запись в данный регистр передает n - ое слово для S преобразования (результат преобразования доступен через 1 такт). Запись в любой из этих регистров выбирает ячейки таблицы для изменения (см. <i>регистр изменения таблицы</i>). Чтение из данного регистра возвращает результат S преобразования n-го слова.	RW

Регистр изменения таблицы $TABLE_CHANGE$ (адрес: $0x20$)

Бит	Сокращенное наименование	Функция	Доступ
7:0	TABLE_VAL_0	Запись в данное поле изменяет значение преобразования 0 - го (младшего) байта входного слова (адрес записи выбирается записью значение в $TRANSFORM_n$). Чтение данного поля возвращает адрес изменяемой ячейки таблицы.	RW
15:8	TABLE_VAL_1	Запись в данное поле изменяет значение преобразования 1 - го байта входного слова (адрес записи выбирается записью значение в $TRANSFORM_n$). Чтение данного поля возвращает адрес изменяемой ячейки таблицы.	RW
23:16	TABLE_VAL_2	Запись в данное поле изменяет значение преобразования 2 - го байта входного слова (адрес записи выбирается записью значение в $TRANSFORM_n$). Чтение данного поля возвращает адрес изменяемой ячейки таблицы.	RW
31:24	TABLE_VAL_3	Запись в данное поле изменяет значение преобразования 3 - го (старшего) байта входного слова (адрес записи выбирается записью значение в $TRANSFORM_n$). Чтение данного поля возвращает адрес изменяемой ячейки таблицы.	RW

Инициализация 8 битного S-BOX

- В регистр TRANSFORM_0 задать значение 0x00_00_00_00
- В регистр TABLE_CHANGE задать значение 0xTT_TT_TT_TT, где TT значение преобразования, 0x00 → TT
- В регистр TRANSFORM_0 задать значение 0x01_01_01_01
- В регистр TABLE_CHANGE задать значение 0xTT_TT_TT_TT, где TT значение преобразования, 0x01 → TT
- ...
- В регистр TRANSFORM_0 задать значение 0xFF_FF_FF_FF
- В регистр TABLE_CHANGE задать значение 0xTT_TT_TT_TT, где TT значение преобразования, 0xFF → TT

Инициализация 4 битного S-BOX

- В регистр TRANSFORM_0 задать значение 0x00_00_00_00
- В регистр TABLE_CHANGE задать значение 0xTP_TP_TP_TP, где Т, Р значение преобразования, 0x0 → Т, 0x0 → Р
- В регистр TRANSFORM_0 задать значение 0x01_01_01_01
- В регистр TABLE_CHANGE задать значение 0xTP_TP_TP_TP, где Т, Р значение преобразования, 0x0 → Т, 0x1 → Р
- ...
- В регистр TRANSFORM_0 задать значение 0x10_10_10_10
- В регистр TABLE_CHANGE задать значение 0xTP_TP_TP_TP, где Т, Р значение преобразования, 0x1 → Т, 0x0 → Р
- ...
- В регистр TRANSFORM_0 задать значение 0xFF_FF_FF_FF
- В регистр TABLE_CHANGE задать значение 0xTP_TP_TP_TP, где Т, Р значение преобразования, 0xF → Т, 0xF → Р

Выполнение преобразования

В регистры TRANSFORM_0 ... TRANSFORM_7 записать необходимое число слов для преобразования, затем считать преобразованное значение из регистров TRANSFORM_0 ... TRANSFORM_7. В случае преобразования одного слова (преобразование 32 бит) между записью в регистр TRANSFORM_0 и чтением результата из этого регистра необходима пауза в 1 такт. В случае преобразования более 32 бит, чтение можно выполнять сразу же после записи всех регистров. Для преобразования можно использовать любое подмножество регистров, результаты преобразования будут в регистрах с соответствующими номерами.

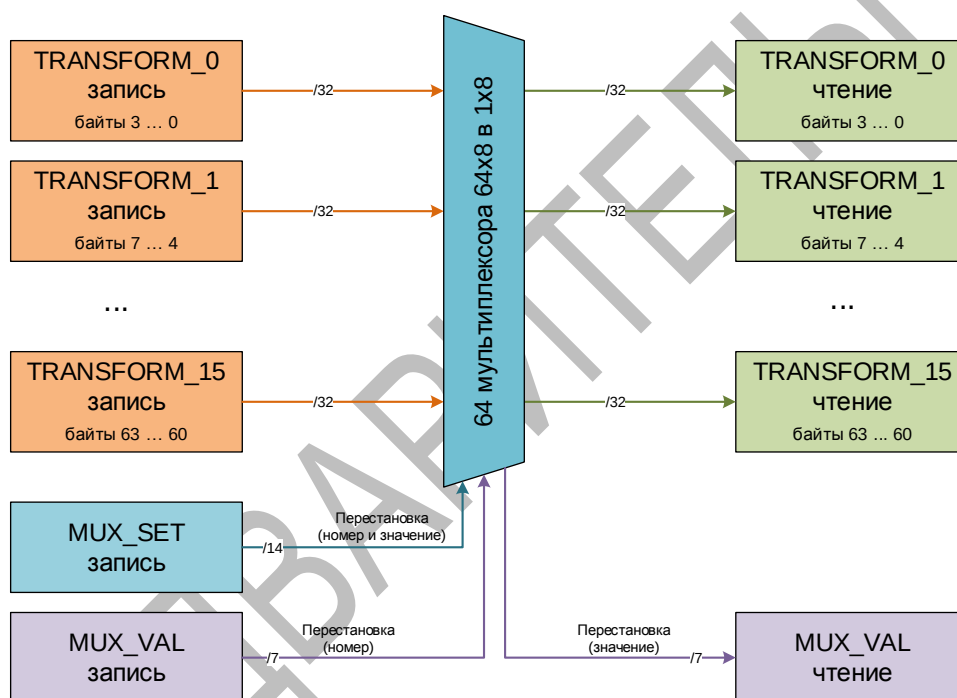
Блок байтовой замены p-byte

Блок осуществляет перестановку байт входного 64 байтового массива

Особенности

- Настраиваемая таблица перестановки (каждому выходному байту задается номер используемого входного байта)
- Обновление выходного массива данных за такт после изменения входного массива
- Конвейерный прием и обработка до 16 слов размером 32 бита
- АНВ подключение

Рисунок 104. Структурная схема блока p-byte.



Регистры модуля

Базовый Адрес	Название	Описание
0x6002_8000	P_BYTE	Блок байтовой замены p-byte
Смещение		
0x00	TRANSFORM_0	Регистр задания слова данных для перестановки 1
0x04	TRANSFORM_1	Регистр задания слова данных для перестановки 2
0x08	TRANSFORM_2	Регистр задания слова данных для перестановки 3
0x0C	TRANSFORM_3	Регистр задания слова данных для перестановки 4
0x10	TRANSFORM_4	Регистр задания слова данных для перестановки 5

0x14	TRANSFORM_5	Регистр задания слова данных для перестановки 6
0x18	TRANSFORM_6	Регистр задания слова данных для перестановки 7
0x1C	TRANSFORM_7	Регистр задания слова данных для перестановки 8
0x20	TRANSFORM_8	Регистр задания слова данных для перестановки 9
0x24	TRANSFORM_9	Регистр задания слова данных для перестановки 10
0x28	TRANSFORM_10	Регистр задания слова данных для перестановки 11
0x2C	TRANSFORM_11	Регистр задания слова данных для перестановки 12
0x30	TRANSFORM_12	Регистр задания слова данных для перестановки 13
0x34	TRANSFORM_13	Регистр задания слова данных для перестановки 14
0x38	TRANSFORM_14	Регистр задания слова данных для перестановки 15
0x3C	TRANSFORM_15	Регистр задания слова данных для перестановки 16
0x40	MUX_SET	Регистр задания перестановки
0x44	MUX_VAL	Регистр чтения перестановки

Регистр задания n – ого слова данных для перестановки TRANSFORM_ n (адрес: $0x00 + 4 * n \dots 0x3C$)

Бит	Сокращенное наименование	Функция	Доступ
31:0	DATA	Запись в данный регистр передает n - ое слово для перестановки. Чтение из данного регистра возвращает n - ое слово результата перестановки.	RW

Регистр задания перестановки MUX_SET (адрес: 0x40)

При записи в данный регистр (младшие 16 бит требуется записывать одновременно) задается перестановка. Выходному байту с номером OUT_SEL ставится в соответствие входной байт с номером IN_SEL.

Бит	Сокращенное наименование	Функция	Доступ
5:0	IN_SEL	Запись задает перестановку IN_SEL $\rightarrow$ OUT_SEL Данное поле только для записи, при чтении возвращается 0	WO

Бит	Сокращенное наименование	Функция	Доступ
7:6	-	Зарезервировано	RO
13:8	OUT_SEL	Запись задает перестановку IN_SEL → OUT_SEL Данное поле только для записи, при чтении возвращается 0	WO
31:14	-	Зарезервировано	RO

Регистр чтения перестановки MUX_VAL (адрес: 0x44)

Бит	Сокращенное наименование	Функция	доступ
5:0	VAL	Данное поле только для чтения, запись игнорируется. При чтении из данного поля возвращается значение перестановки для выбранного через SEL выходного байта.	RO
7:6	-	Зарезервировано	RO
13:8	SEL	Запись в данное поле задает выходной байт, для которого буде возвращено значение перестановки при чтении регистра. Данное поле только для записи, при чтении возвращается 0.	WO
31:14	-	Зарезервировано	RO

Инициализация

- В регистр MUX_SET задать значение 0x0000_00_TT, где TT номер входного байта, который необходимо выдавать на 0 байт
- В регистр MUX_SET задать значение 0x0000_01_TT, где TT номер входного бита, который необходимо выдавать на 1 байт
- ...
- В регистр MUX_SET задать значение 0x0000_3F_TT, где TT номер входного бита, который необходимо выдавать на 63 байт

Выполнение преобразования

В регистры TRANSFORM_0 ... TRANSFORM_15 записать данные для преобразования. Сразу после этого из регистров TRANSFORM_0 ... TRANSFORM_15 можно считать результат перестановки. Обновление выходных данных происходит одновременно с записью любой части входных данных.

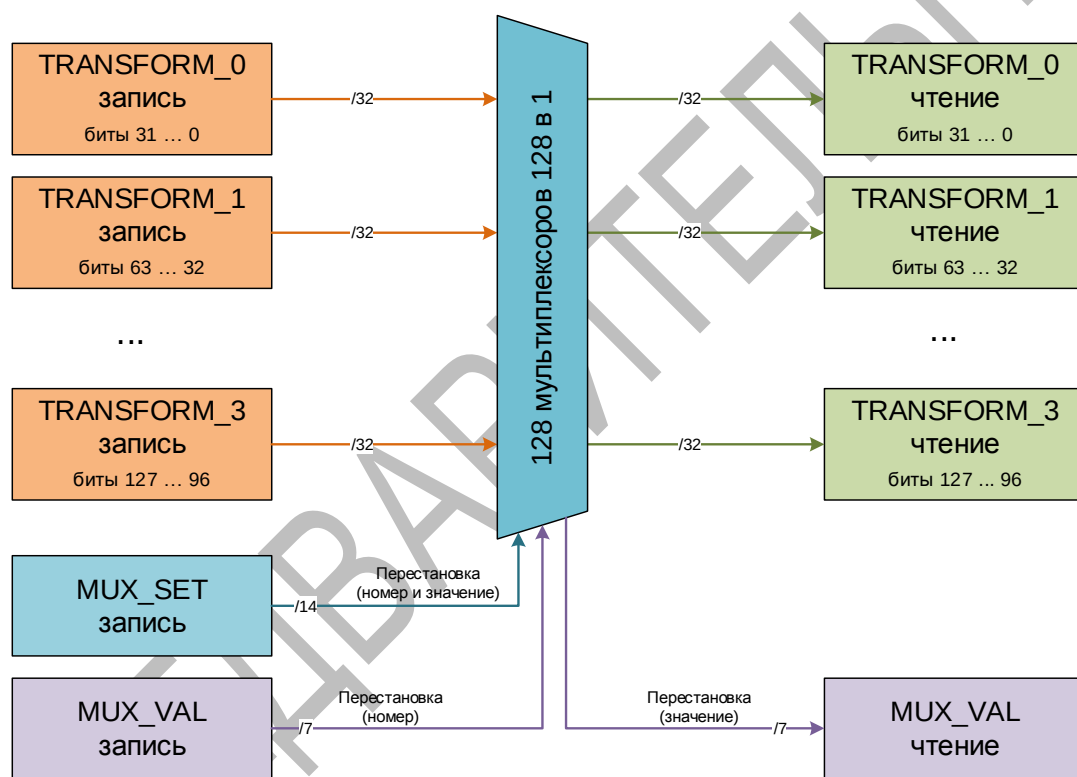
Блок битовой замены p-bit

Блок выполняет перестановку по таблице бит входного 128 битного слова.

Особенности

- Возможность задания таблицы перестановки бит (каждому выходному биту задается номер используемого входного бита)
- Обновление выходного слова в том же такте после изменения входного
- Конвейерный прием и обработка до четырех 32 битных слов
- АНВ подключение

Рисунок 105. Структурная схема блока p-bit.



Регистры модуля

Базовый Адрес	Название	Описание
0x6002_C000	P_BIT0	Блок битовой замены p-bit
0x6003_0000	P_BIT1	Блок битовой замены p-bit
0x6003_4000	P_BIT2	Блок битовой замены p-bit
0x6003_8000	P_BIT3	Блок битовой замены p-bit
Смещение		
0x00	TRANSFORM_0	Регистр задания слова данных для перестановки 1
0x04	TRANSFORM_1	Регистр задания слова данных для перестановки 2
0x08	TRANSFORM_2	Регистр задания слова данных

		для перестановки 3
0x0C	TRANSFORM_3	Регистр задания слова данных для перестановки 4
0x10	MUX_SET	Регистр задания перестановки
0x14	MUX_VAL	Регистр чтения перестановки

Регистр задания n – ого слова данных для перестановки $TRANSFORM_n$ (адрес: $0x00 + 4 * n \dots 0x0C$)

Бит	Сокращенное наименование	Функция	Доступ
31:0	DATA	Запись в данный регистр передает n - ое слово для перестановки. Чтение из данного регистра возвращает n - ое слово результата перестановки.	RW

Регистр задания перестановки MUX_SET (адрес: $0x10$)

При записи в данный регистр (младшие 16 бит требуется записывать одновременно) задается перестановка. Выходному биту с номером OUT_SEL ставится в соответствие входной бит с номером IN_SEL .

Бит	Сокращенное наименование	Функция	Доступ
6:0	IN_SEL	Запись задает перестановку $IN_SEL \rightarrow OUT_SEL$. Данное поле только для записи, при чтении возвращается 0.	WO
7	-	Зарезервировано	RO
14:8	OUT_SEL	Запись задает перестановку $IN_SEL \rightarrow OUT_SEL$. Данное поле только для записи, при чтении возвращается 0.	WO
31:15	-	Зарезервировано	RO

Регистр чтения перестановки MUX_VAL (адрес: $0x14$)

Бит	Сокращенное наименование	Функция	Доступ
6:0	VAL	Данное поле только для чтения, запись игнорируется. При чтении из данного поля возвращается значение перестановки для выбранного через SEL выходного бита.	RO
7	-	Зарезервировано	RO
14:8	SEL	Запись в данное поле задает выходной бит, для которого буде возвращено значение перестановки при чтении регистра. Данное поле только для записи, при чтении возвращается 0.	WO

Бит	Сокращенное наименование	Функция	Доступ
31:15	-	Зарезервировано	RO

Инициализация

- В регистр MUX_SET задать значение 0x0000_00_TT, где TT номер входного бита, который необходимо выдавать на 0 бит
- В регистр MUX_SET задать значение 0x0000_01_TT, где TT номер входного бита, который необходимо выдавать на 1 бит
- ...
- В регистр MUX_SET задать значение 0x0000_7F_TT, где TT номер входного бита, который необходимо выдавать на 127 бит

Выполнение преобразования

В регистры TRANSFORM_0 ... TRANSFORM_3 записать данные для преобразования. Сразу после этого из регистров TRANSFORM_0 ... TRANSFORM_3 можно считать результат перестановки. Обновление выходных данных происходит одновременно с записью любой части входных данных.

Модуль обработки датчиков безопасности (SENSORS)

Общее описание

Блок подключается к шине APB, размер данных 32 бита, размер адреса 5. Модуль учитывает возможность доступа на запись по 32, 16 и 8 бит.

Блок используется для обработки датчиков обнаружения атак на защищенное ядро. Модуль обрабатывает плавающую сетку. Плавающую сетку модуль периодически подключает к нулю и питанию, и контролирует что данный уровень появился на сетке (нарушение сетки или замыкание ее на другие сетки должно привести к расхождению заданного и принятого уровней).

Так же блок имеет дополнительные входы для обработки датчиков ЭМИ и света. Условная схема подключения представлена на рисунке ниже.

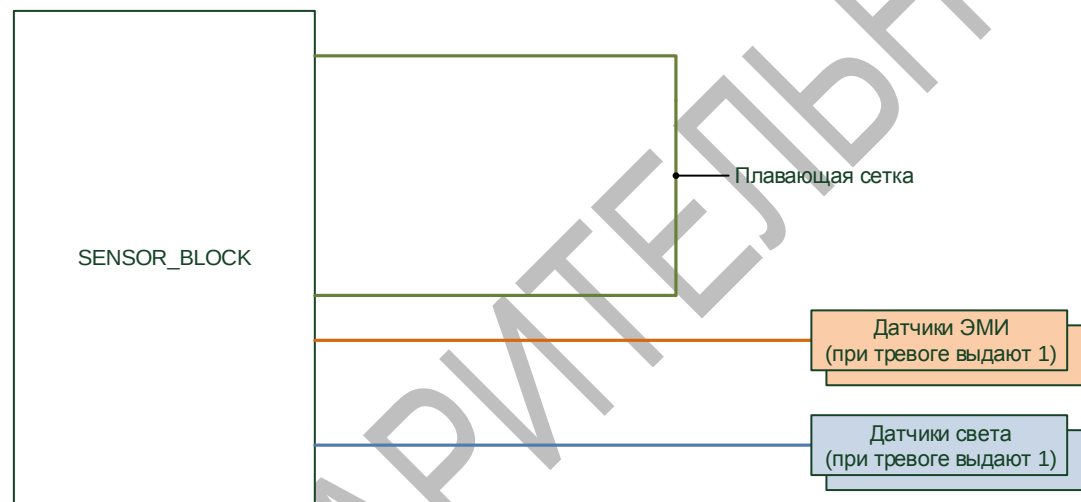


Рисунок 106. Условная схема подключения блока.

В состав модуля входит регистр статуса, где отмечается срабатывание датчиков. Регистр, для непрерывного мониторинга состояния выходов сенсоров. Регистр для настройки интервала переключения плавающей сетки. Также 2 регистра для настройки сигналов прерывания и запроса сброса памяти ключей.

Описание функционирования блока и режимов работы

Блок непрерывно мониторит состояние входов датчиков. Регистр отражает значение входов плавающей защитной сетки, а также дополнительных датчиков ЭМИ и света (`sens_alarms[0]`-`sens_alarms[4]`).

Блок отвечает за срабатывание входов плавающей защитной сетки, а также дополнительных датчиков ЭМИ и света (`sens_alarms[0]`-`sens_alarms[4]`).

При наличии ненулевого входа на `sens_alarms[0]`-`sens_alarms[3]`, в биты 0-3 регистра статуса **state_reg** записывается 1. Через заданный промежуток времени (в последнем такте перед переключением), при отличающихся уровнях между входом и выходом плавающей сетки (`sens_alarms[4]=1`), в бит 4 регистра статуса **state_reg** записывается 1. Сброс флага срабатывания дополнительных датчиков производится записью 0 в биты 0-3 **state_reg**, только при условии отсутствия ненулевого входа. Сброс флага срабатывания датчика для плавающей сетки производится записью 0 в бит 4 **state_reg**, при записи в момент проверки (конец временного интервала, заданного в регистре `switch_counter`), сброс произойдет только

при условии совпадения уровней входа и выхода. Запись 0 в середине интервала переключения сетки, сбросит флаг срабатывания до ближайшей проверки. Запись 1 в биты регистра не меняет его состояния.

Модуль SENSORS имеет регистр настройки интервала переключения плавающей сетки. Время переключения задается в тактах рабочей частоты в диапазоне от 3 до 2^8 .

Модуль SENSORS имеет специальные регистры настройки с выбором датчиков вызывающих формирование сигнала запроса прерывания и сигнала запроса сброса памяти ключей. Сигнал запроса прерывания и сброса формируется при наличии в **status_reg** единичных битов, помеченных маской в регистрах **int_mask** и **k_res_mask** соответственно.

Карта регистров

Таблица ниже содержит названия и адреса всех программно-доступных регистров блока SENSORS. Поле Таблицы «Адрес» указывает относительный адрес регистра в шестнадцатеричной форме. Поле Таблицы «Доступ» указывает на политику доступа к заданному регистру: r/w – чтение и запись, r/o – только чтение (запись не имеет эффекта).

Таблица. Карта регистров SENSORS.

Название регистра	Адрес	Доступ	Общее описание регистра
state_reg	0x00	r/w	Регистр статуса → сигналы с датчиков
real_time	0x04	r/o	Регистр мониторинга состояния входов сенсоров
int_mask	0x0C	r/w	Регистр маски запроса прерывания
k_res_mask	0x10	r/w	Регистр маски запроса сброса
enable	0x18	r/w	Регистр включения датчиков
mesh_key7	0x1C	r/w	Регистр ключа защитной сетки 0 (биты 255:224)
mesh_key6	0x20	r/w	Регистр ключа защитной сетки 1 (биты 223:192)
mesh_key5	0x24	r/w	Регистр ключа защитной сетки 2 (биты 191:160)
mesh_key4	0x28	r/w	Регистр ключа защитной сетки 3 (биты 159:128)
mesh_key3	0x2C	r/w	Регистр ключа защитной сетки 4 (биты 127:96)
mesh_key2	0x30	r/w	Регистр ключа защитной сетки 5 (биты 95:64)
mesh_key1	0x34	r/w	Регистр ключа защитной сетки 6 (биты 63:32)
mesh_key0	0x38	r/w	Регистр ключа защитной сетки 7 (биты 31:0)
mesh_init1	0x3C	r/w	Регистр инициализации данных защитной сетки 0 (биты 63:32)
mesh_init0	0x40	r/w	Регистр инициализации данных защитной сетки 1

			(биты 31:0)
meshcntr	0x44	r/w	Регистр управления защитной сетки

Регистр state_reg

Относительный адрес: 0x00.

Доступ: r/w

Значение после сброса: 0x0.

Регистр статуса **state_reg** предназначен для отображения срабатывания датчиков.

Таблица. Регистр state

Разряды	Назначения разрядов
31:5	Зарезервировано
4	сигнал атаки на защитную сетку - mesh active alarm
3	сигнал датчика ЭМИ 1 - mesh emfi alarm1
2	сигнал датчика ЭМИ 2 mesh emfi alarm2
1	сигнал датчика ЭМИ 3 mesh emfi alarm3
0	сигнал датчика света

Регистр real_time

Относительный адрес: 0x04.

Доступ: r/o.

Значение после сброса: 0x0.

Регистр **real_time** предназначен для непрерывного мониторинга состояния входов сенсоров.

Таблица. Регистр realtime

Разряды	Назначения разрядов
31:5	Зарезервировано
4	сигнал атаки на защитную сетку - mesh active alarm
3	сигнал датчика ЭМИ 1 - mesh emfi alarm1
2	сигнал датчика ЭМИ 2 mesh emfi alarm2
1	сигнал датчика ЭМИ 3 mesh emfi alarm3
0	сигнал датчика света

Регистр int_mask

Относительный адрес: 0x0C.

Доступ: r/w.

Значение после сброса: 0x0.

Регистр **int_mask** предназначен для выбора датчиков, вызывающих формирования сигнала запроса прерывания. Разряды распределены также, как в регистре **state_reg**. Сигнал прерывания формируется при наличии единичных значений в регистре статуса (**state_reg**) в битах, помеченных единицами в регистре маски (**int_mask**). Возможна только однократная запись в регистр.

Регистр **k_res_mask**

Относительный адрес: 0x10.

Доступ: r/w.

Значение после сброса: 0x0.

Регистр **k_res_mask** предназначен для выбора датчиков, вызывающих формирования сигнала запроса сброса памяти ключей. Разряды распределены также, как в регистре **state_reg** (Таблица 3). Сигнал прерывания формируется при наличии единичных значений в регистре статуса (**state_reg**) в битах, помеченных единицами в регистре маски (**key_res_mask**). Возможна только однократная запись в регистр.

Регистр **enable**

Относительный адрес: 0x18.

Доступ: r/w.

Значение после сброса: *все датчики выключены*

Регистр **enable** предназначен для включения-выключения датчиков безопасности, соответствие бит датчикам такое же как в регистре **state_reg**. Единица в бите соответствующего датчика - датчик включен, ноль - датчик выключен. Возможна только однократная запись в регистр.

В ОТП выделено слово, которое читается загрузочной программой и однократно устанавливаются соответствующие биты в регистре.

Таблица. Регистр *enable*

Разряды	Назначения разрядов
31:5	Зарезервировано
4	разрешение mesh active alarm
3	разрешение mesh emfi alarm1
2	разрешение mesh emfi alarm2
1	разрешение mesh emfi alarm3
0	разрешение датчика света

Регистр meshcntr

Относительный адрес: 0x44.

Доступ: r/w.

Значение после сброса: 0x0.

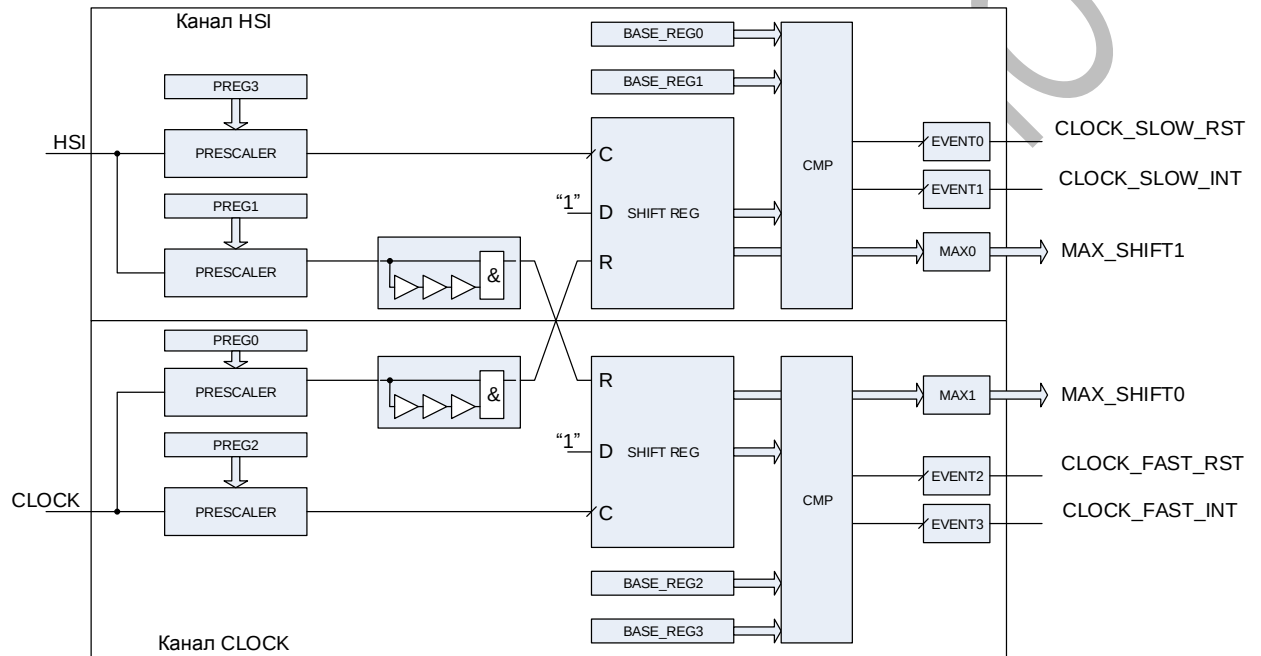
Таблица. Регистр meshcntr

Разряды	Назначения разрядов
31:3	Зарезервировано
2:0	Регистр делителя опорной частоты для работы сетки 000 переключение каждые 3 такта 001 переключение каждые 4 такта 010 переключение каждые 8 тактов 011 переключение каждые 16 такта 100 переключение каждые 32 такта 101 переключение каждые 64 такта 110 переключение каждые 128 такта 111 переключение каждые 256 тактов

Датчик частоты (CLK_MEASURE)

Блок датчика частоты является важным элементом защиты защищенной части микроконтроллера. Для избежания возможных атак через изменение тактовой частоты реализован датчик частоты, который сравнивает подаваемый синхросигнал с опорным, и генерирует один из двух типов событий: сброс ключей батарейного домена, прерывание, в случае существенного отличия синхросигналов. Необходимо обеспечить невозможность отключения синхросигнала HSI.

Рисунок 107. Структурная схема блока датчика частоты.



При настройке блока, исходя из ожидаемых значений частот сигналов HSI и CLOCK, необходимо задать значения предварительных делителей в регистрах $PREGx$ так, чтобы:

$$\begin{cases} \frac{f_{HSI}}{PREG3 + 1} = K_0, \text{ где } K_0 \in [1..15] \\ \frac{f_{CLOCK}}{PREG0 *} \\ \frac{f_{CLOCK}}{PREG2 + 1} = K_1, \text{ где } K_1 \in [1..15] \\ \frac{f_{HSI}}{PREG1 *} \end{cases}$$

* $PREG0$ – при $f_{HSI} > f_{CLOCK}$,
 $PREG0 + 1$ – при $f_{HSI} < f_{CLOCK}$,

* $PREG1$ – при $f_{HSI} < f_{CLOCK}$,
 $PREG1 + 1$ – при $f_{HSI} > f_{CLOCK}$.

Тогда в регистре $SHIFT_REG$ канала HSI будут возникать значения от 0 до $2^{K_0}-1$, а в канале CLOCK – от 0 до $2^{K_1}-1$.

При снижении частоты CLOCK:

- в регистре канала HSI будут возникать значения от 0 до $2^{K_2}-1$, причем $K_2 > K_0$;
- в регистре канала CLOCK будут возникать значения от 0 до $2^{K_3}-1$, причем

$$K_3 < K_1.$$

Если значение $2^{K_2}-1$ превысит или равно значению $BASE_REG0=2^{K_0}-1$, то возникнет событие EVENT0, а если $2^{K_2}-1$ превысит или равно значению $BASE_REG1=2^{K_0}-1$, то возникнет событие EVENT1.

$$\begin{cases} f_{CLOCK} \leq \frac{f_{HSI} \times PREG0 *}{(PREG3 + 1) \times (K_0 + 1)} \Rightarrow EVENT0 \\ f_{CLOCK} \leq \frac{f_{HSI} \times PREG0 *}{(PREG3 + 1) \times (K_0 + 1)} \Rightarrow EVENT1 \end{cases}$$

При увеличении частоты CLOCK:

- в регистре канала HSI будет возникать значения от 0 до $2^{K_2}-1$, причем $K_2 < K_0$;
- в регистре канала CLOCK будет возникать значения от 0 до $2^{K_3}-1$, причем $K_3 > K_1$.

Если значение $2^{K_3}-1$ превысит или равно значению $BASE_REG2=2^{K_1}-1$, то возникнет событие EVENT2, если $2^{K_3}-1$ превысит или равно значению $BASE_REG3=2^{K_1}-1$, то возникнет событие EVENT3.

$$\begin{cases} f_{CLOCK} \geq \frac{f_{HSI} \times (PREG2 + 1) \times (K_1 + 1)}{PREG1 * } \Rightarrow EVENT2 \\ f_{CLOCK} \geq \frac{f_{HSI} \times (PREG2 + 1) \times (K_1 + 1)}{PREG1 * } \Rightarrow EVENT3 \end{cases}$$

Для иллюстрации принципов работы блока возможные комбинации настроек приведены в таблице ниже.

Частота HSI	Ожидаемая частота CLOCK	Реальная частота CLOCK	PREG3	PREG1	PREG0	PREG2	BASE REG0	BASE REG1	BASE REG2	BASE REG3	Прим
8 МГц	60 МГц	60 МГц	1	4	29	9	4	4	8	8	K0 = 2 K1 = 3 Нет событий
8 МГц	60 МГц	80 МГц	1	4	29	9	4	4	8	8	K0 = 2 K1 = 3 K2 = 1.5 K3 = 4 EVENT2 EVENT3 Высокая частота
8 МГц	60 МГц	30 МГц	1	4	29	9	4	4	8	8	K0 = 2 K1 = 3 K2 = 4 K3 = 1,5

											EVENT0 EVENT1 Низкая частота
--	--	--	--	--	--	--	--	--	--	--	---------------------------------------

Также можно программно считать максимальное значение регистра *SHIFT_REG* каждого из каналов, которое он достигал. При чтении регистра *SHIFT_REG* считывается теневой регистр, в котором отражается максимальное достигнутое регистром *SHIFT_REG* значение за период с последнего сброса теневого регистра. Сброс теневого регистра осуществляется программно, записью управляющего бита EN=0.

Карта регистров

Базовый Адрес		Название	Описание
0x400B_0000		CLK_MEASURE	Блок датчика частоты
Смещение			
0x0000	0	CLK_CNTR_STAT	Регистр настроек и состояния
0x0004	1	ALARM_SHIFT_RST	Регистр задания пороговых значений для сигнала сброса
0x0008	2	ALARM_SHIFT_INT	Регистр задания пороговых значений для сигнала прерывания
0x000c	3	ALARM_PREG_0	Регистр настройки параметров сброса счетчиков
0x0010	4	ALARM_PREG_1	Регистр настройки параметров сброса счетчиков
0x0014	5	CLK_STAT	Регистр максимального значения SHIFT_REG каждого из каналов

CLK_CNTR_STAT

Бит	Имя	Значение	Описание
15...6	Зарезервировано		Зарезервировано, читается как 0
5	KEY_RESET		Флаг возникновения события сброса ключей
4	INTERRUPT		Флаг возникновения прерывания
3...1	Зарезервировано		Зарезервировано, читается как 0
0	EN		Бит разрешения работы блока контроля частоты 0 – блок выключен 1 – блок включен

ALARM_SHIFT_RST

Бит	Имя	Значение	Описание
31...16	BASE_REG2		Значение MAX_SHIFT0, при котором возникает событие сброса ключей
15...0	BASE_REG0		Значение MAX_SHIFT1, при котором возникает событие сброса ключей

ALARM_SHIFT_INT

Бит	Имя	Значение	Описание
31...16	BASE_REG3		Значение MAX_SHIFT0, при котором возникает прерывание
15...0	BASE_REG1		Значение MAX_SHIFT1, при котором возникает прерывание

ALARM_PREG_0

Бит	Имя	Значение	Описание
31...16	PREG0		Поле основания счета для счетчика 0. Всегда задавать больше нуля.
15...0	PREG1		Поле основания счета для счетчика 1. Всегда задавать больше нуля.

ALARM_PREG_1

Бит	Имя	Значение	Описание
31...16	PREG2		Поле основания счета для счетчика 2. Всегда задавать больше нуля.

15...0	PREG3		Поле основания счета для счетчика 3. Всегда задавать больше нуля.
--------	-------	--	--

Генератор случайных чисел (RANDOM)

Назначение и свойства

Модуль служит для управления двумя кольцевыми генераторами, их включения, тестирования, а также формирования случайного 32-х битного числа.

- Модуль содержит два генератора.
- Модуль позволяет включать и выключать генераторы группами по 1.
- Модуль позволяет задать паузу после включения генераторов, перед началом сбора случайного числа.
- Модуль позволяет включать генераторы по биту регистра управления.
- Модуль имеет несколько режимов запуска сбора случайного числа.
 - одиночный
 - постоянный

Модуль формирует сигнал прерывания по окончании сбора случайного числа.

Датчик случайных чисел представляет собой механизм получения непредсказуемой последовательности чисел, основанный на физическом недетерминированном процессе.

Случайные числа становятся доступны пользователям поблочно и только после успешного прохождения блоком первичных статистических тестов (критерий серий и критерий Хи-квадрат). В случае, если последовательность не проходит статистическое тестирование, она должна быть удалена и должна быть сгенерирована новая последовательность.

Для увеличения количества случайных чисел и улучшения их статистических характеристик, пользователь может воспользоваться аппаратно-программным механизмом генерации псевдослучайных чисел. В этом случае выход генератора случайных чисел является рандомизирующим фактором для выполнения операции шифрования с использованием стойкого шифра.

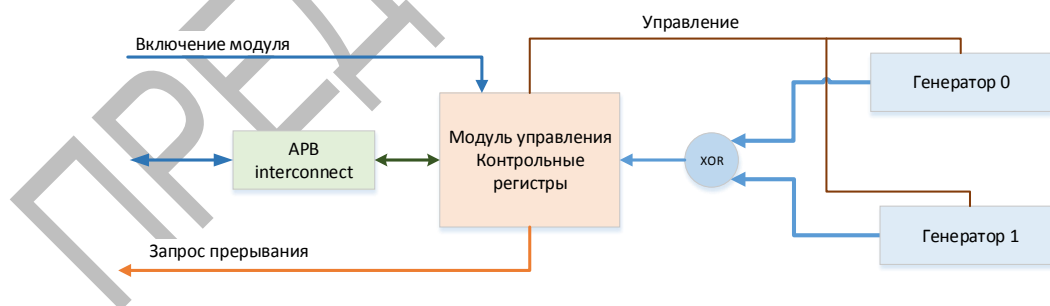


Рисунок 108. Структурная схема блока генератора случайных чисел.

Регистры модуля

Базовый Адрес	Название	Описание
---------------	----------	----------

0x400B_8000	RANDOM	Блок генератора случайных чисел
Смещение		
0x00	STAT_CTRL_REG	Регистр статуса и управления
0x04	INT_CTRL_REG	Регистр управления прерыванием
0x08	CLK_DIV_REG	Регистр делителя синхросигнала генератора
0x0C	PAUSE_REG	Регистр паузы включения
0x10	OUTPUT_REG	Регистр случайного значения
0x14	PAUSE_CNT_REG	Регистр счётчика паузы
0x18	TEMP_REG	Регистр сбора случайного числа

Регистр статуса и управления STAT_CTRL_REG (адрес: 0x00), начальное значение 0x000000F0

Бит	Сокращенное наименование	Функция	Доступ
0	FORCE_EN	Бит принудительного включения генераторов, при установленном данном бите в 1.	RW
1	EN_STATE	Генераторы включены, данный бит устанавливается в 1 если есть хоть одна группа включенных генераторов (значение MASK не нулевое или бит FORCE_EN=1) и вышла пауза после включения (см. PAUSE_REG)	RO
2	BUSY	Генератор занят, идет сбор случайного числа	RO
3	UNREAD_DATA	Флаг наличия несчитанных данных. После сбора очередного случайного числа в данном бите появляется 1, бит сбрасывается при чтении случайного числа (см. OUTPUT_REG)	RO
5:4	MASK	Маска выбора активного генератора: 4 бит – генератор 0, 5 бит – генератор 1.	RW
7:6	-	Зарезервировано	RO
9:8	WORK_MODE	Режим сбора случайного числа 00 - сбор случайных чисел остановлен 01 - однократный запуск сбора случайного числа (после начала сбора поле сбрасывается в 00 автоматически) 10 - сбор нового случайного числа начинается если нет непрочитанного случайного числа (UNREAD_DATA равен 0)	RW

		11 - постоянный сбор случайных чисел, сбор нового случайного числа начинается сразу после окончания сбора очередного числа	
11:10	-	Зарезервировано	RO
17:12	BIT_CN	Счетчик оставшихся бит до сбора нового случайного числа	RO
31:18	-	Зарезервировано	RO

Регистр управления прерыванием INT_CTRL_REG (адрес: 0x04)

Бит	Сокращенное наименование	Функция	Доступ
0	INT_EN	Разрешение прерывания. При наличии 1 в этом бите и 1 в поле UNREAD_DATA , на выходе interrupt появляется 1.	RW
1	UNREAD_DATA	Флаг наличия несчитанных данных. После сбора очередного случайного числа в данном бите появляется 1, бит сбрасывается при чтении случайного числа (см. OUTPUT_REG)	RO
31:2	-	Зарезервировано	RO

Регистр делителя блока генератора CLK_DIV_REG (адрес: 0x08), начальное значение 7 (500 КГц @ 8МГц)

Бит	Сокращенное наименование	Функция	Доступ
15:0	DIV	Для формирования блока защелки данных генератора используется входной блок модуля, деленный на $2 * (DIV + 1)$. По описанию модуля генераторов частота защелки данных на генераторах должна быть порядка 500 КГц.	RW
31:16	-	Зарезервировано	RO

Регистр паузы включения PAUSE_REG (адрес: 0x0C)

Бит	Сокращенное наименование	Функция	Доступ
31:0	PAUSE	После включения генераторов через внешний сигнал или бит управляющего регистра, при включении любого блока через маску, при добавлении новых включенных блоков к работающим, сбор случайного числа может быть начат не ранее чем выйдет данная пауза. Если включение происходит во время сбора случайного числа, сбор приостанавливается до тех пор, пока не выйдет данная пауза.	RW

Регистр случайного значения OUTPUT_REG (адрес: 0x10)

Бит	Сокращенное наименование	Функция	Доступ
31:0	DATA	Последнее собранное случайного число. Во время сбора очередного случайного числа, значение данного регистра неизменно. Обновление данных происходит по окончании сбора случайного числа	RO

Регистр счетчика паузы PAUSE_CNT_REG (адрес: 0x14)

Бит	Сокращенное наименование	Функция	Доступ
31:0	CUR_PAUSE	Текущее значение счетчика паузы включения генераторов (значение для справки). 0 значение означает что пауза выдержана.	RO

Регистр сбора случайного числа TEMP_REG (адрес: 0x18)

Бит	Сокращенное наименование	Функция	Доступ
31:0	TEMP_DATA	Текущее значение сдвигового регистра сбора случайного числа (значение для справок).	RO

Контроллер UART (ISO7816)

Введение

Универсальный синхронно-асинхронный приемопередатчик (USART) предлагает гибкие средства полнодуплексного обмена с внешним оборудованием, требующим стандартного формата кодирования данных NRZ. Модуль USART поддерживает широкий диапазон скорости передачи данных с использованием генератора дробной скорости передачи данных.

Модуль поддерживает синхронную одностороннюю передачу данных и полудуплексную однопроводную передачу данных. Также поддерживается протокол Smartcard, и операции модема (CTS/RTS).

Особенности модуля USART

- Полный дуплекс, асинхронный обмен данными
- Стандартный формат данных NRZ (Mark/Space)
- Конфигурируемый метод передискретизации на 16 или 8 для обеспечения гибкости между скоростью и допуском по тактовой частоте
- Системы генератора дробной скорости передачи данных
 - программируемая скорость передачи и приема данных (значение скорости передачи данных при максимальной частоте шины APB см. в спецификациях).
- Программируемая длина слова данных (8 или 9 бит)
- Конфигурируемые стоп-биты – поддержка 1 или 2 стоп-бит
- Тактовый выход передатчика для синхронной передачи данных
- Возможность эмуляции смарт-карты (режим Smartcard)
 - Интерфейс Smartcard поддерживает асинхронный протокол для смарт-карт в соответствии со стандартом ISO 7816-3
 - стоп-биты 0.5, 1.5 для интерфейса Smartcard
- Однопроводная полудуплексная передача данных
- Отдельные биты разрешения (enable) для передатчика и приемника
- Флаги обнаружения передачи:
 - Буфер приемника полон
 - Буфер передатчика пуст
 - Флаг завершения передачи
- Проверка четности:
 - Передача бита четности
 - Проверка четности принятого байта данных
- Флаги обнаружения ошибки:
 - Переполнение
 - Обнаружение шума
 - Ошибка фрейма
 - Ошибка четности
- Прерывания с флагами:
 - Изменение состояния CTS
 - Регистр данных передатчика пуст
 - Передача завершена
 - Регистр данных приемника заполнен
 - Ошибка переполнения

- Ошибка фрейма
- Обнаружение шума
- Ошибка четности

Функциональное описание USART

Интерфейс подключается к внешним приборам при помощи трех выводов (см. Рисунок 109). Любая двунаправленная передача данных USART требует минимум два вывода: вход для данных (RX) и выход для передачи данных (TX):

RX: вход для последовательных принимаемых данных. Для восстановления данных используется техника передискретизации, чтобы отделить нужные входящие данные от шума.

TX: выход для передачи данных. Когда передатчик запрещен, вывод возвращается в состояние, заданное конфигурацией порта ввода-вывода (I/O).

Когда передатчик разрешен, но никакие данные не передаются, на выводе TX устанавливается высокий уровень (логическая 1). В однопроводном режиме и режиме smart-карты, данный порт ввода-вывода используется для передачи и приема данных (SW_RX).

При помощи данных выводов происходит передача и прием последовательных данных в нормальном режиме работы модуля USART, в виде фреймов, содержащих:

- Сигнал ожидания линии (Idle) до передачи или приема
- Стартовый бит
- Слово данных (8 или 9 бит), MSB передается первым
- 0.5, 1, 1.5, 2 стоп-бита, показывающие, что фрейм завершен
- Данный интерфейс использует дробный генератор скорости передачи данных - с 12-битной мантиссой и 4-битной дробной частью
- Регистр статуса (USART_SR)
- Регистр данных (USART_DR)
- Регистр скорости передачи (USART_BRR) - 12-битная мантисса и 4-битная дробная часть
- Регистр защитного интервала (Guardtime register) (USART_GTPR) в случае использования режима Smartcard.

Описание регистров и их бит приведено в разделе «Регистры USART».

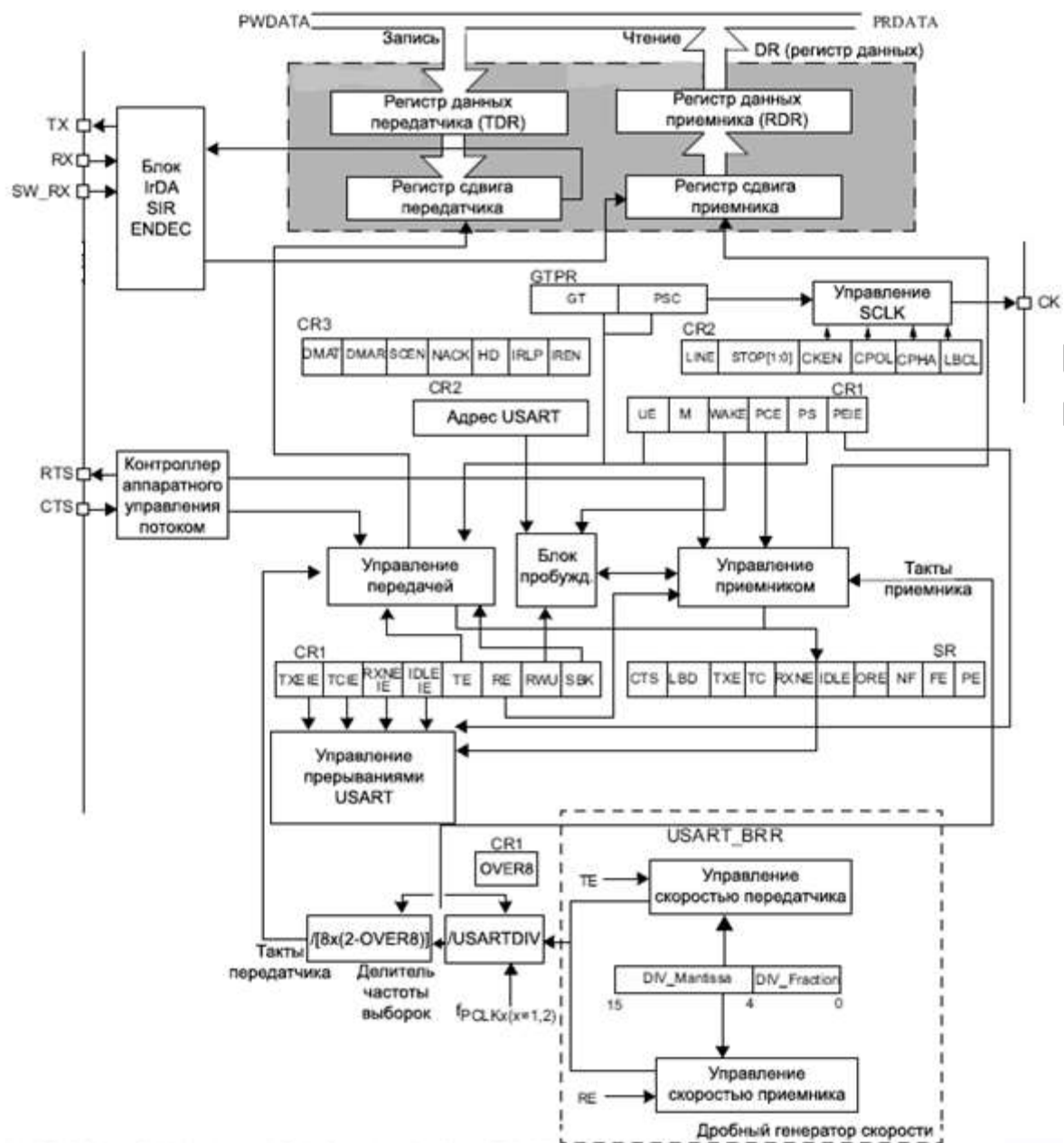
Для работы в синхронном режиме требуется дополнительный вывод:

- **СК:** Выход тактов передатчика. На этот вывод выдаются такты данных для синхронной передачи, соответствующей режиму ведущий SPI (нет тактовых импульсов в стартовых и стоп- битах, и программная опция отправки тактового импульса на последнем бите данных). Параллельно данные могут синхронно приниматься по выводу RX. Это можно использовать для управления внешними периферийными устройствами, у которых есть регистры сдвига (например, LCD драйверы). Фаза и полярность тактов выбирается программно. В режиме Smartcard, вывод СК обеспечивает такты для smart-карты.

В режиме аппаратного управления потоком данных требуются дополнительно выводы:

- **CTS:** сигнал Clear To Send блокирует передачу данных по окончании текущей передачи при высоком уровне сигнала
- **RTS:** сигнал Request to send показывает готовность USART к приему данных (при низком уровне сигнала).

Рисунок 109. Блок схема USART



$$\text{USARTDIV} = \text{DIV_Mantissa} + (\text{DIV_Fraction} / 8 \times (2 - \text{OVER8}))$$

Описание символов USART

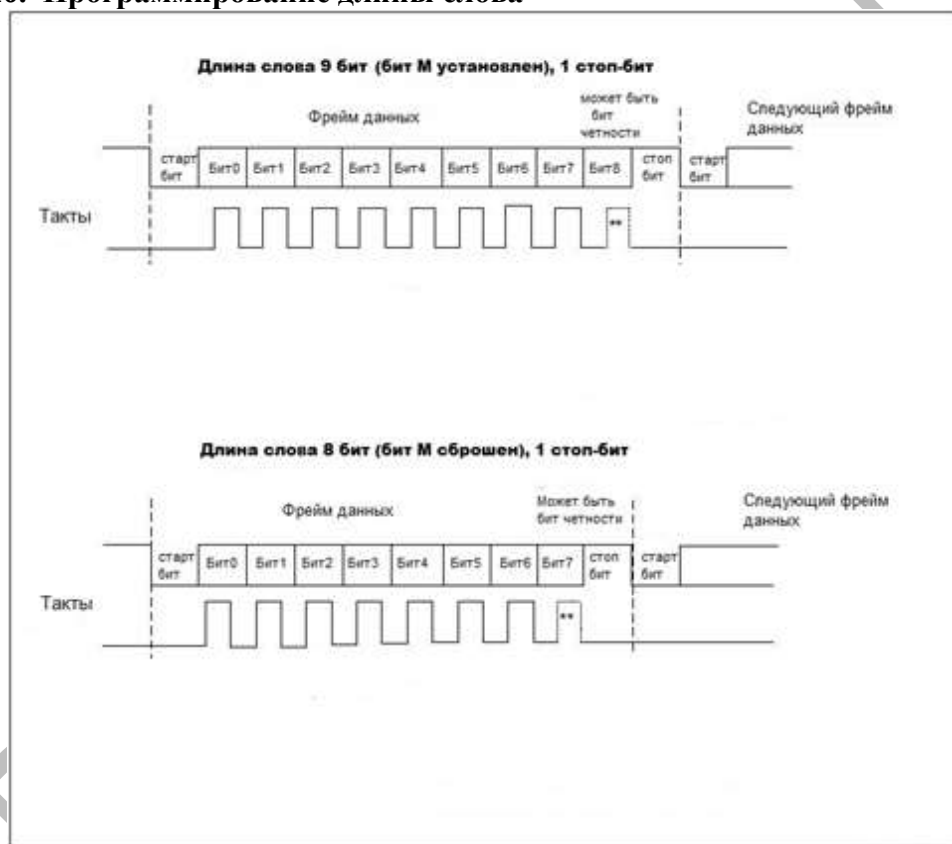
Для передачи можно выбрать длину слова 8 или 9 бит путем программирования бита М в регистре USART_CR1 (см. Рисунок 110).

Вывод TX находится в состоянии логического 0 во время передачи стартового бита, и в состоянии логической 1 во время передачи стоп-бита.

Передача и прием управляются генератором скорости передачи. Такты для передатчика и приемника генерируются, когда установлен соответствующий бит разрешения.

Ниже подробно описан каждый блок.

Рисунок 110. Программирование длины слова



Передатчик

Передатчик может отправлять слова данных по 8 или 9 бит, в зависимости от статуса бита М. Когда установлен бит разрешения передачи (TE), данные из регистра сдвига выводятся на вывод TX, и соответствующие тактовые импульсы выводятся на вывод СК.

Передача символа

Во время передачи USART данные сдвигаются на вывод TX, младший значащий бит (МЗБ) идет первым. В данном режиме, регистр USART_DR состоит из буфера (TDR) между внутренней шиной и регистром сдвига передачи (см. Рисунок 109).

Каждому символу предшествует стартовый бит низкого уровня (логический 0) в течение длительности одного бита. Символ завершается конфигурируемым количеством стоп-битов.

USART модуль поддерживает следующие стоп-биты: 0.5, 1, 1.5 и 2 стоп-бита.

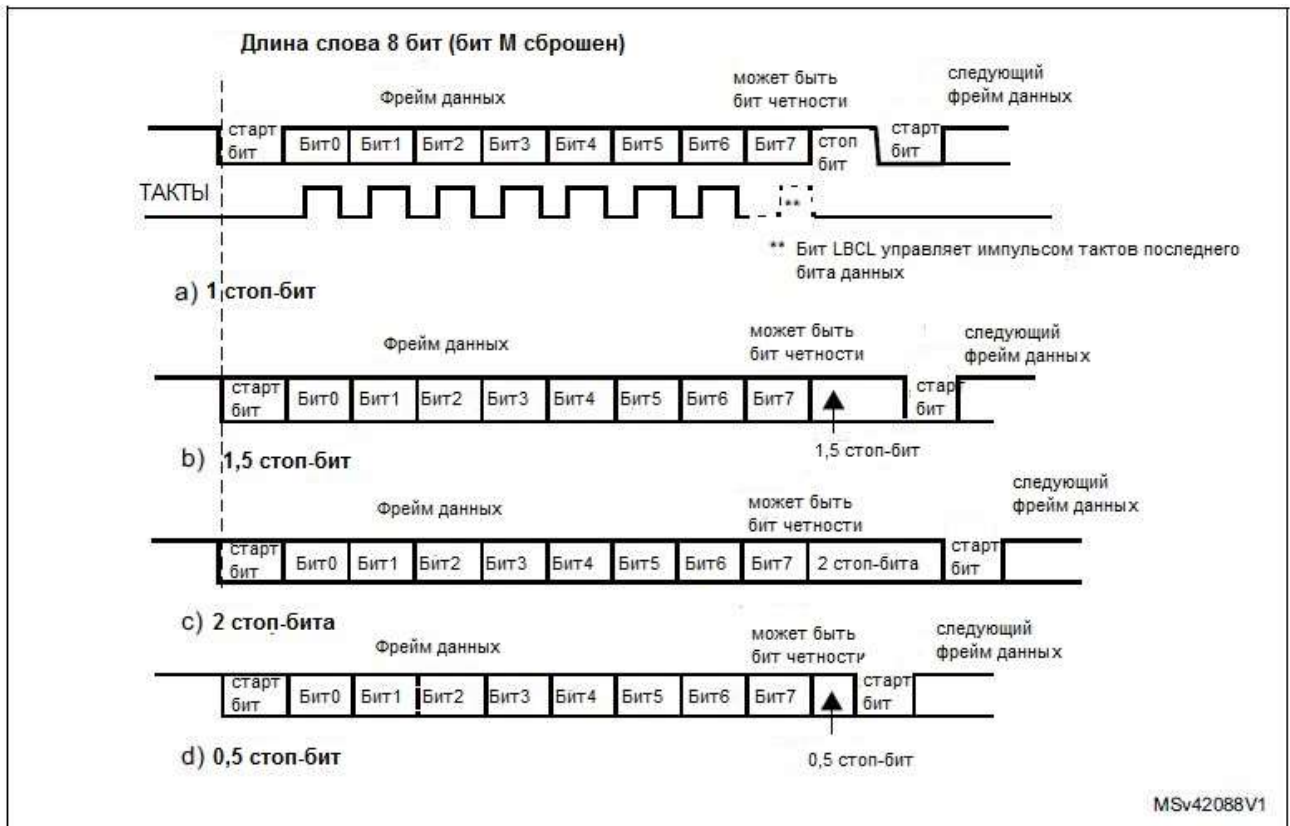
Примечание: бит TE не должен сбрасываться во время передачи данных. Сброс бита TE во время передачи приведет к повреждению данных на выводе TX, так как счетчики генератора скорости остановятся. Текущие передаваемые данные будут потеряны.

После разрешения бита TE будет отправлен фрейм idle.

Конфигурируемые стоп-биты

Количество передаваемых стоп-бит с каждым символом может быть запрограммировано в Регистре Управления 2 битами 13,12.

- **1 стоп-бит:** значение по умолчанию.
- **2 стоп-бита:** Это значение поддерживается в обычном режиме USART, однопроводном режиме и режиме модема.
- **0.5 стоп-бит:** для использования при приеме данных в режиме Smartcard.
- **1.5 стоп-бит:** для использования при передаче и приеме данных в режиме Smartcard.



Процедура программирования:

1. Разрешить USART записью в 1 бита UE в регистре USART_CR1.
2. Запрограммировать бит М в регистре USART_CR1, чтобы определить длину слова данных.
3. Запрограммировать количество стоп-бит в регистре USART_CR2.
4. Выбрать необходимую скорость передачи при помощи регистра USART_BRR.
5. Установить бит TE в регистре USART_CR1.
6. Записать данные для передачи в регистр USART_DR (это сбросит бит TXE). Повторить данную операцию для каждого передаваемого символа в случае использования одиночного буфера.
7. После записи последних данных в регистр USART_DR, подождать, пока TC станет равным 1. Это покажет, что передача последнего фрейма завершена. Например, это требуется, когда модуль USART отключен или переходит в режиме Halt, чтобы избежать повреждения последней передачи.

Однобайтный обмен

Бит TXE всегда сбрасывается при записи в регистр данных.

Бит TXE устанавливается аппаратно, и указывает, что:

- Данные были перемещены из регистра TDR в регистр сдвига, и началась передача.
- Регистр TDR пуст.
- Следующие данные могут быть записаны в регистр USART_DR без риска перезаписи предыдущих данных.

Данный флаг генерирует прерывание, если установлен бит TXE.

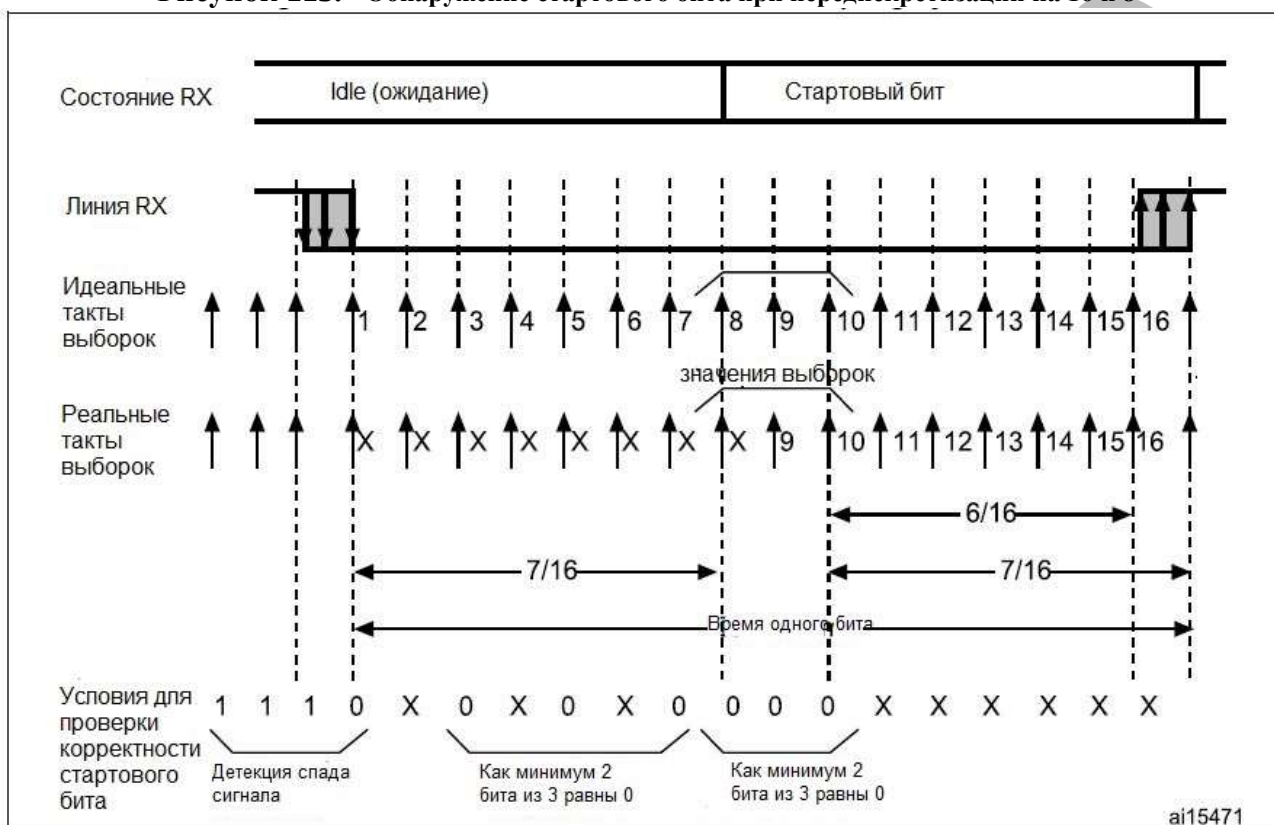
USART может принимать слово данных разрядностью 8 или 9 бит, в зависимости от значения бита M в регистре USART_CR1.

Обнаружение стартового бита

Последовательность обнаружения стартового бита одинаковая при передискретизации сигнала с кратностью 16 или 8.

В приемопередатчике USART стартовый бит детектируется, когда распознана определенная последовательность выборки сигнала: 1 1 1 0 X 0 X 0 X 0 0 0 0.

Рисунок 113. Обнаружение стартового бита при передискретизации на 16 и 8



Примечание: Если последовательность не завершена, обнаружение стартового бита прекращается, и приемник переходит в состояние ожидания (idle) (никакие флаги не установлены) и ожидает заднего фронта.

Стартовый бит подтвержден (флаг RXNE установлен, генерируется прерывание при RXNEIE=1), если 3 бита выборки равны 0 (первая выборка на 3, 5 и 7 бите находит три бита в логическом 0, а вторая выборка на 8, 9 и 10 битах также находит три бита в 0).

Стартовый бит подтвержден (флаг RXNE установлен, генерируется прерывание при RXNEIE=1), но устанавливается флаг NE, если при обеих выборках как минимум 2 из 3 бит установлены в 0 (выборка на 3, 5 и 7 битах и выборка на 8, 9 и 10 битах). Если данное условие не выполняется, не происходит обнаружение стартового бита, и приемник возвращается в состояние ожидания (idle) (нет установленных флагов).

Если при одной из выборок (выборка на 3, 5 и 7 битах или выборка на 8, 9 и 10 битах) 2 из 3 бит находятся в логическом 0, стартовый бит подтверждается, но устанавливается флаг NE.

Прием символа

При приеме данных данные перемещаются на вывод RX, начиная с MSB. В данном режиме регистр

USART_DR состоит из буфера (RDR) между внутренней шиной и регистром сдвига приемника.

Процедура программирования:

1. Разрешить USART путем записи в 1 бита UE в регистре USART_CR1.
2. Запрограммировать бит M в регистре USART_CR1 для определения длины слова данных.
3. Запрограммировать количество стоп-бит в регистре USART_CR2.
5. Выбрать необходимую скорость передачи данных при помощи регистра USART_BRR
6. Установить бит RE в регистре USART_CR1. Это активирует приемник, и он начнет обнаружение стартового бита.

Когда принят символ

- Устанавливается бит RXNE. Он показывает, что содержимое сдвигового регистра передано в RDR. Другими словами, данные были получены и могут быть прочитаны (а также связанные с ними флаги ошибок).
- Генерируется прерывание, если установлен бит RXNEIE.
- Флаг ошибки устанавливается, если во время приема была обнаружена ошибка фрейма, шума или переполнения.
- В мультибуферном режиме бит RXNE устанавливается после каждого принятого байта и сбрасывается чтением в регистре данных.
- В режиме одиночного буфера бит RXNE сбрасывается программно при чтении регистра USART_DR. Флаг RXNE сбрасывается записью 0. Бит RXNE должен быть сброшен перед окончанием приема следующего символа, чтобы избежать ошибки переполнения.

Примечание: бит RE не должен сбрасываться при приеме данных. Если бит RE запрещен во время приема, прием текущего байта будет прерван.

Ошибка переполнения (overflow)

Ошибка переполнения возникает, когда получен символ, но бит RXNE не сброшен. Данные не могут быть переданы из регистра сдвига в регистр RDR, пока бит RXNE не будет сброшен.

Флаг RXNE устанавливается после каждого полученного байта. Ошибка переполнения возникает, если флаг RXNE установлен, когда приняты следующие данные. При появлении ошибки переполнения:

- Бит ORE будет установлен.
- Содержимое регистра RDR не будет потеряно. Предыдущие данные будут доступны при чтении регистра USART_DR.

- Регистр сдвига будет перезаписан. В этот момент любые данные полученные во время ошибки переполнения будут потеряны.
- Генерируется прерывание, если либо установлены биты RXNEIE или EIE.
- Бит ORE сбрасывается чтением регистра USART_SR, за которым следует операция чтения регистра USART_DR.

Примечание: установленный бит ORE показывает, что как минимум один элемент данных потерян. Существует два варианта:

- если RXNE=1, то последние корректные данные сохраняются в регистр RDR и могут быть прочитаны,
- если RXNE=0, это означает, что последние корректные данные уже были прочитаны, поэтому нечего считывать из регистра RDR. Это может произойти, когда последние корректные данные были прочитаны из регистра RDR одновременно с получением новых данных. Также это может произойти, когда новые данные приняты во время последовательности чтения (между доступом на чтение регистра USART_SR и доступом на чтение регистра USART_DR).

Выбор правильного метода передискретизации

Приемник использует разные техники передискретизации, конфигурируемые пользователем (кроме синхронного режима), для восстановления данных путем отделения друг от друга входящих данных и шума.

Метод передискретизации можно выбрать путем программирования бита OVER8 в регистре USART_CR1, передискретизация может быть или 16- или 8-кратная тактам скорости (Рисунок 114 и Рисунок 115).

В зависимости от применения:

- Выберите передискретизацию на 8 (OVER8=1) для достижения увеличенной скорости (до fPCLK/8). В данном случае максимальный допуск на отклонение тактов для приемника снижается (см. Допуск отклонения тактовой частоты USART приемника);
- Выберите передискретизацию на 16 (OVER8=0) для увеличения допуска приемника к отклонению тактов. В данном случае максимальная скорость ограничена fPCLK/16.

Программирование бита ONEBIT в регистре USART_CR3 выбирает метод, используемый для оценки логического уровня. Есть две опции:

- мажоритарная выборка из 3 выборок по центру принятого бита. В данном случае, если эти 3 выборки не одинаковые, установится бит NF;
- одна выборка в центре принятого бита.

В зависимости от применения:

– выберите метод мажоритарности из 3 выборок (ONEBIT=0) при работе в условиях шума и отклоните данные, если обнаружен шум, потому что это показывает, что во время выборки произошел сбой.

– выберите метод одиночной выборки (ONEBIT=1), если нет шумов на линии, чтобы увеличить допуск на отклонение тактов приемника (см. Допуск на отклонения тактовой частоты USART приемника). В данном случае бит NF не будет установлен.

Когда во фрейме обнаружен шум:

- Бит NF будет установлен по переднему фронту бита RXNE.
- Некорректные данные будут перемещены из регистра сдвига в регистр USART_DR.
- Не будет генерироваться прерывание в случае однобайтной передачи данных. Однако, этот бит будет установлен одновременно с битом RXNE, который сам генерирует прерывание. В случае мультибуферного обмена данными прерывание возникнет, если установлен бит EIE в регистре USART_CR3.

Бит NF сбрасывается чтением регистра USART_SR, после чего следует чтение регистра USART_DR.

Примечание: передискретизация на 8 недоступна в режимах Smartcard. В данном режиме бит OVER8 аппаратно сбрасывается в 0.

Рисунок 114. Выборка данных при передискретизации на 16

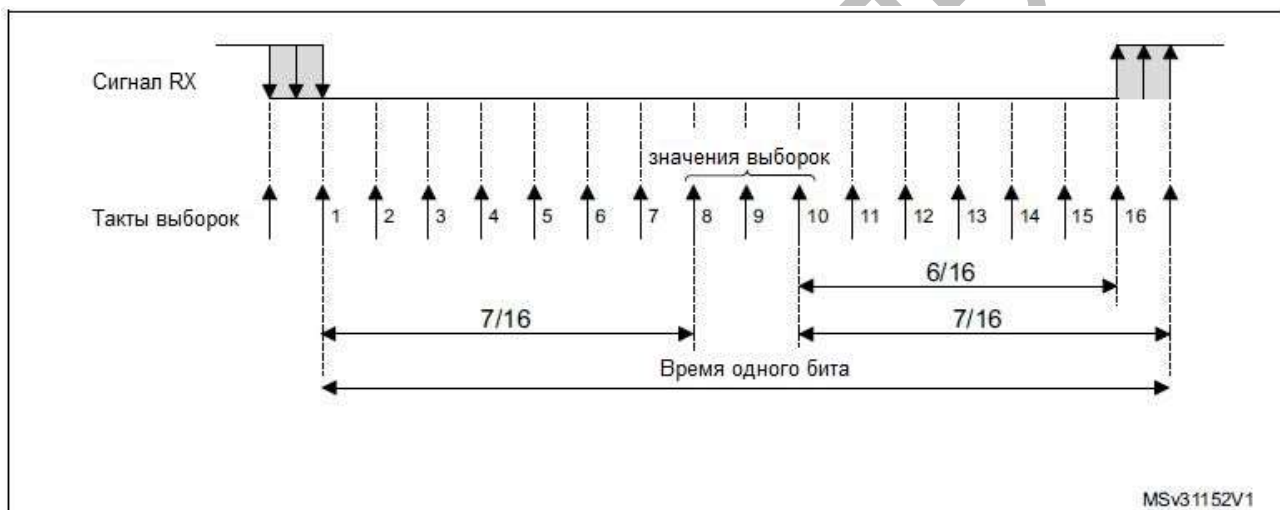


Рисунок 115. Выборка данных при передискретизации на 8

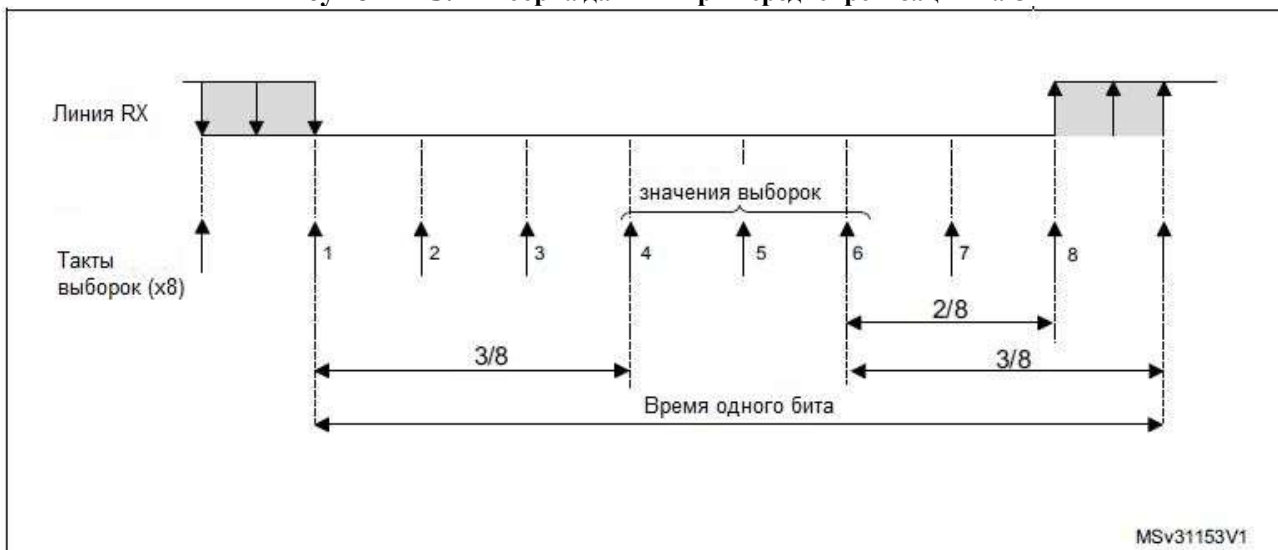


Таблица. Обнаружение шумов для считанных данных

Значение	NE статус	Значение принятого бита
000	0	0
001	1	0
010	1	0
011	1	1
100	1	0
101	1	1
110	1	1
111	0	1

Ошибка фрейма (framing error)

Ошибка фрейма обнаруживается, когда:

Стоп-бит не распознан во время приема данных в ожидаемый промежуток времени из-за рассинхронизации или из-за чрезмерного шума.

Когда обнаружена ошибка фрейма:

- Бит FE устанавливается аппаратно
- Некорректные данные передаются из регистра сдвига в регистр USART_DR.
- В случае однобайтного обмена никакое прерывание не генерируется. Однако, данный бит устанавливается одновременно с битом RXNE, который сам генерирует прерывание. В случае мультибуферного обмена прерывание генерируется, если установлен бит EIE в регистре USART_CR3.

Бит FE сбрасывается чтением регистра USART_SR, после чего следует чтение регистра USART_DR.

Конфигурируемые стоп-биты во время приема

Количество принимаемых стоп-бит можно конфигурировать в Регистре Управления 2. Это может быть 1 или 2 бита в нормальном режиме и 0.5 или 1.5 в режиме Smartcard.

1. **0.5 стоп-бит (прием в режиме Smartcard)**: для данного режиме выборки не делаются. Как следствие, при выборе 0.5 стоп-бита ошибка фрейма или сигнал break не могут быть обнаружены.

2. **1 стоп-бит**: анализ сигнала для 1 стоп-бита осуществляется на 8-ой, 9-ой и 10-й выборке.

3. **1.5 стоп-бита (режим Smartcard)**: при передаче данных в режиме smartcard, прибор должен проверить корректность отправляемых данных. Таким образом, блок приемника должен быть разрешен (RE=1 в регистре USART_CR1), а стоп-бит проверяется на наличие ошибки четности. В случае наличия ошибки четности, смарт-карта подтягивает сигнал данных в логический 0 во время выборки (сигнал NACK), который опознается как ошибка фрейма. После чего флаг FE устанавливается с RXNE по окончании 1.5 стоп-бита. Анализ уровня для 1.5 стоп-бита осуществляется на 16, 17 и 18 выборках (1 период скорости после начала стоп-бита). 1.5 стоп-бит может быть разложен на 2 части: одна часть из 0.5 периода скорости, во время которой ничего не происходит, и вторая часть – 1 период нормального стоп-бита, во время которого происходит анализ сигнала. См. Режим Smartcard.

4. **2 стоп-бита**: Анализ для 2 стоп-бит осуществляется на 8, 9 и 10 выборках первого стоп-бита. Если была обнаружена ошибка фрейма во время первого стоп-бита, устанавливается флаг ошибки. Второй стоп-бит не проверяется на наличие ошибки фрейма. Флаг RXNE устанавливается по окончании первого стоп-бита.

Дробный генератор скорости

Скорость обмена для приемника и передатчика (Rx и Tx) устанавливается в одинаковое значение, программируемое коэффициентами Mantissa (целая часть) и Fraction (дробная часть) делителя USARTDIV.

Формула 1: Скорость для стандартного USART (включая режим SPI)

$$\text{Tx/Rx baud} = \frac{f_{\text{СК}}}{8 \times (2 - \text{OVER8}) \times \text{USARTDIV}}$$

Формула 2: Скорость для режимов Smartcard

$$\text{Tx/Rx baud} = \frac{f_{\text{СК}}}{16 \times \text{USARTDIV}}$$

USARTDIV – это число с фиксированной запятой без знака, запрограммированное в регистре USART_BRR.

- Если OVER8=0, дробная часть кодируется 4 битами и программируется битами DIV_fraction [3:0] в регистре USART_BRR
- Если OVER8=1, дробная часть кодируется 3 битами и программируется битами DIV_fraction [2:0] в регистре USART_BRR, бит DIV_fraction [3] должен быть сброшен.

Примечание: Счетчики скорости обновляются в регистрах скорости после операции записи в регистр USART_BRR. Следовательно, значение регистра скорости не должно изменяться во время активного обмена данными.

Примеры расчета USARTDIV для значений регистра USART_BRR при OVER8=0

Пример 1:

Если DIV_Mantissa = 0d27 и DIV_Fraction = 0d12 (USART_BRR = 0x1BC), тогда
Mantissa (USARTDIV) = 0d27
Fraction (USARTDIV) = $12/16 = 0d0.75$
Следовательно, USARTDIV = 0d27.75

Пример 2:

Для программирования USARTDIV = 0d25.62
получается:
DIV_Fraction = $16 * 0d0.62 = 0d9.92$
Ближайшее действительное число 0d10 = 0xA
DIV_Mantissa = mantissa (0d25.620) = 0d25 = 0x19
Тогда, USART_BRR = 0x19A, следовательно, USARTDIV = 0d25.625

Пример 3:

Для программирования USARTDIV = 0d50.99
получается:
DIV_Fraction = $16 * 0d0.99 = 0d15.84$
Ближайшее действительное число 0d16 = 0x10 => переполнение DIV_frac[3:0] => должен быть добавлен перенос (carry) к мантиссе
DIV_Mantissa = mantissa (0d50.990 + carry) = 0d51 = 0x33
Тогда, USART_BRR = 0x330, следовательно, USARTDIV = 0d51.000

Примеры расчета USARTDIV для значений регистра USART_BRR при OVER8=1

Пример 1:

Если DIV_Mantissa = 0x27 и DIV_Fraction[2:0] = 0d6 (USART_BRR = 0x1B6), тогда
Mantissa (USARTDIV) = 0d27
Fraction (USARTDIV) = $6/8 = 0d0.75$
Therefore USARTDIV = 0d27.75

Пример 2:

Для программирования USARTDIV = 0d25.62
получается:
DIV_Fraction = $8 * 0d0.62 = 0d4.96$
Ближайшее действительное число 0d5 = 0x5
DIV_Mantissa = mantissa (0d25.620) = 0d25 = 0x19
Тогда, USART_BRR = 0x195 => USARTDIV = 0d25.625

Пример 3:

Для программирования USARTDIV = 0d50.99

Получается:

$$\text{DIV_Fraction} = 8 * 0d0.99 = 0d7.92$$

Ближайшее действительное число $0d8 = 0x8 \Rightarrow$ переполнение $\text{DIV_frac}[2:0] \Rightarrow$ должен быть добавлен перенос (carry) к мантиссе

$$\text{DIV_Mantissa} = \text{mantissa} (0d50.990 + \text{carry}) = 0d51 = 0x33$$

$$\text{Тогда, USART_BRR} = 0x0330 \Rightarrow \text{USARTDIV} = 0d51.000$$

Таблица. Определение погрешности для запрограммированных скоростей передачи при $f_{\text{PCLK}} = 8 \text{ МГц}$ или $f_{\text{PCLK}} = 12 \text{ МГц}$, передискретизация на 16(1)

Передискретизация на 16 (OVER8=0)							
Скорость передачи		$f_{\text{PCLK}} = 8 \text{ МГц}$			$f_{\text{PCLK}} = 12 \text{ МГц}$		
№	Требуемая	Реальная	Значение, запрограммированное в регистре скорости передачи	% погрешности = (Вычисленное значение – Требуемое значение) x скорость передачи / желаемую скорость передачи	Реальная	Значение, запрограммированное в регистре скорости передачи	% погрешности
1	1.2 Кбит/с	1.2 Кбит/с	416.6875	0	1.2 Кбит/с	625	0
2	2.4 Кбит/с	2.4 Кбит/с	208.3125	0.01	2.4 Кбит/с	312.5	0
3	9.6 Кбит/с	9.604 Кбит/с	52.0625	0.04	9.6 Кбит/с	78.125	0
4	19.2 Кбит/с	19.185 Кбит/с	26.0625	0.08	19.2 Кбит/с	39.0625	0
5	38.4 Кбит/с	38.462 Кбит/с	13	0.16	38.339 Кбит/с	19.5625	0.16
6	57.6 Кбит/с	57.554 Кбит/с	8.6875	0.08	57.692 Кбит/с	13	0.16
7	115.2 Кбит/с	115.942 Кбит/с	4.3125	0.64	115.385 Кбит/с	6.5	0.16
8	230.4 Кбит/с	228.571 Кбит/с	2.1875	0.79	230.769 Кбит/с	3.25	0.16
9	460.8 Кбит/с	470.588 Кбит/с	1.0625	2.12	461.538 Кбит/с	1.625	0.16
10	921.6 Кбит/с	NA	NA	NA	NA	NA	NA
11	2 Мбит/с	NA	NA	NA	NA	NA	NA
12	3 Мбит/с	NA	NA	NA	NA	NA	NA

1. Чем ниже тактовая частота центрального процессора, тем меньше точность для конкретного значения скорости передачи. Верхний предел достижимой скорости передачи может быть зафиксирован с этими данными.

Таблица. Определение погрешности для запрограммированных скоростей передачи при $f_{\text{PCLK}} = 8 \text{ МГц}$ или $f_{\text{PCLK}} = 12 \text{ МГц}$, передискретизация на 8(1)

Передискретизация на 8 (OVER8=0)							
Скорость передачи		$f_{\text{PCLK}} = 8 \text{ МГц}$			$f_{\text{PCLK}} = 12 \text{ МГц}$		
№	Требуемая	Реальная	Значение, запрограммированное в регистре скорости передачи	% погрешности = (Вычисленное значение – Требуемое значение) x скорость передачи / желаемую скорость передачи	Реальная	Значение, запрограммированное в регистре скорости передачи	% погрешности
1	1.2 Кбит/с	1.2 Кбит/с	833.375	0	1.2 Кбит/с	1250	0
2	2.4 Кбит/с	2.4 Кбит/с	416.625	0.01	2.4 Кбит/с	625	0
3	9.6 Кбит/с	9.604 Кбит/с	104.125	0.04	9.6 Кбит/с	156.25	0
4	19.2 Кбит/с	19.185 Кбит/с	52.125	0.08	19.2 Кбит/с	78.125	0
5	38.4 Кбит/с	38.462 Кбит/с	26	0.16	38.339 Кбит/с	39.125	0.16
6	57.6 Кбит/с	57.554 Кбит/с	17.375	0.08	57.692 Кбит/с	26	0.16
7	115.2 Кбит/с	115.942 Кбит/с	8.625	0.64	115.385 Кбит/с	13	0.16
8	230.4 Кбит/с	228.571 Кбит/с	4.375	0.79	230.769 Кбит/с	6.5	0.16
9	460.8 Кбит/с	470.588 Кбит/с	2.125	2.12	461.538 Кбит/с	3.25	0.16
10	921.6 Кбит/с	888.889 Кбит/с	1.125	3.55	923.077 Кбит/с	1.625	0.16
11	2 Мбит/с	NA	NA	NA	NA	NA	NA
12	3 Мбит/с	NA	NA	NA	NA	NA	NA

1. Чем ниже тактовая частота центрального процессора, тем меньше точность для конкретного значения скорости передачи. Верхний предел достижимой скорости передачи может быть зафиксирован с этими данными.

**Таблица. Определение погрешности для запрограммированных скоростей передачи при $f_{PCLK} = 16$ МГц
или $f_{PCLK} = 24$ МГц,
передискретизация на 16(1)**

Передискретизация на 16 (OVER8=0)							
Скорость передачи		$f_{PCLK} = 16$ МГц			$f_{PCLK} = 24$ МГц		
№	Требуемая	Реальная	Значение, запрограммированное в регистре скорости передачи	% погрешности = (Вычисленное значение – Требуемое значение) x скорость передачи / желаемую скорость передачи	Реальная	Значение, запрограммированное в регистре скорости передачи	% погрешности
1	1.2 Кбит/с	1.2 Кбит/с	833.3125	0	1.2	1250	0
2	2.4 Кбит/с	2.4 Кбит/с	416.6875	0	2.4	625	0
3	9.6 Кбит/с	9.598 Кбит/с	104.1875	0.02	9.6	156.25	0
4	19.2 Кбит/с	19.208 Кбит/с	52.0625	0.04	19.2	78.125	0
5	38.4 Кбит/с	38.369 Кбит/с	26.0625	0.08	38.4	39.0625	0
6	57.6 Кбит/с	57.554 Кбит/с	17.375	0.08	57.554	26.0625	0.08

**Таблица. Определение погрешности для запрограммированных скоростей передачи при $f_{PCLK} = 16$ МГц
или $f_{PCLK} = 24$ МГц,
передискретизация на 16(1). (продолжение)**

Передискретизация на 16 (OVER8=0)							
Скорость передачи		$f_{PCLK} = 16$ МГц			$f_{PCLK} = 24$ МГц		
№	Требуемая	Реальная	Значение, запрограммированное в регистре скорости передачи	% погрешности = (Вычисленное значение – Требуемое значение) x скорость передачи / желаемую скорость передачи	Реальная	Значение, запрограммированное в регистре скорости передачи	% погрешности
7	115.2 Кбит/с	115.108 Кбит/с	8.6875	0.08	115.385	13	0.16
8	230.4 Кбит/с	231.884 Кбит/с	4.3125	0.64	230.769	6.5	0.16
9	460.8 Кбит/с	457.143 Кбит/с	2.1875	0.79	461.538	3.25	0.16
10	921.6 Кбит/с	941.176 Кбит/с	1.0625	2.12	923.077	1.625	0.16
11	2 Мбит/с	NA	NA	NA	NA	NA	NA
12	3 Мбит/с	NA	NA	NA	NA	NA	NA

1. Чем ниже тактовая частота центрального процессора, тем меньше точность для конкретного значения скорости передачи. Верхний предел достижимой скорости передачи может быть зафиксирован с этими данными.

**Таблица. Определение погрешности для запрограммированных скоростей передачи при $f_{PCLK} = 16$ МГц
или $f_{PCLK} = 24$ МГц,
передискретизация на 8(1)**

Передискретизация на 8 (OVER8=0)							
Скорость передачи		$f_{PCLK} = 16$ МГц			$f_{PCLK} = 24$ МГц		
№	Требуемая	Реальная	Значение, запрограммированное в регистре скорости передачи	% погрешности = (Вычисленное значение – Требуемое значение) x скорость передачи / желаемую скорость передачи	Реальная	Значение, запрограммированное в регистре скорости передачи	% погрешности
1	1.2 Кбит/с	1.2 Кбит/с	1666.625	0	1.2 Кбит/с	2500	0
2	2.4 Кбит/с	2.4 Кбит/с	833.375	0	2.4 Кбит/с	1250	0
3	9.6 Кбит/с	9.598 Кбит/с	208.375	0.02	9.6 Кбит/с	312.5	0
4	19.2 Кбит/с	19.208 Кбит/с	104.125	0.04	19.2 Кбит/с	156.25	0
5	38.4 Кбит/с	38.369 Кбит/с	52.125	0.08	38.4 Кбит/с	78.125	0
6	57.6 Кбит/с	57.554 Кбит/с	34.75	0.08	57.554 Кбит/с	52.125	0.08
7	115.2 Кбит/с	115.108 Кбит/с	17.375	0.08	115.385 Кбит/с	26	0.16
8	230.4 Кбит/с	231.884 Кбит/с	8.625	0.64	230.769 Кбит/с	13	0.16
9	460.8 Кбит/с	457.143 Кбит/с	4.375	0.79	461.538 Кбит/с	6.5	0.16
10	921.6 Кбит/с	941.176 Кбит/с	2.125	2.12	923.077 Кбит/с	3.25	0.16
11	2 Мбит/с	2000 Кбит/с	1	0	2000 Кбит/с	1.5	0
12	3 Мбит/с	NA	NA	NA	3000 Кбит/с	1	0

1. Чем ниже тактовая частота центрального процессора, тем меньше точность для конкретного значения скорости передачи. Верхний предел достижимой скорости передачи может быть зафиксирован с этими данными.

**Таблица. Определение погрешности для запрограммированных скоростей передачи при $f_{PCLK} = 8$ МГц
или $f_{PCLK} = 16$ МГц,
передискретизация на 16(1)**

Передискретизация на 16 (OVER8=0)							
Скорость передачи		$f_{PCLK} = 8$ МГц			$f_{PCLK} = 16$ МГц		
№	Требуемая	Реальная	Значение, запрограммированное в регистре скорости передачи	% погрешности = (Вычисленное значение – Требуемое значение) x скорость передачи / желаемую скорость передачи	Реальная	Значение, запрограммированное в регистре скорости передачи	% погрешности
1	2.4 Кбит/с	2.400 Кбит/с	208.3125	0.00%	2.400 Кбит/с	416.6875	0.00%
2	9.6 Кбит/с	9.604 Кбит/с	52.0625	0.04%	9.598 Кбит/с	104.1875	0.02%
3	19.2 Кбит/с	19.185 Кбит/с	26.0625	0.08%	19.208 Кбит/с	52.0625	0.04%
4	57.6 Кбит/с	57.554 Кбит/с	8.6875	0.08%	57.554 Кбит/с	17.3750	0.08%
5	115.2 Кбит/с	115.942 Кбит/с	4.3125	0.64%	115.108 Кбит/с	8.6875	0.08%
6	230.4 Кбит/с	228.571 Кбит/с	2.1875	0.79%	231.884 Кбит/с	4.3125	0.64%
7	460.8 Кбит/с	470.588 Кбит/с	1.0625	2.12%	457.143 Кбит/с	2.1875	0.79%
8	896 Кбит/с	NA	NA	NA	888.889 Кбит/с	1.1250	0.79%
9	921.6 Кбит/с	NA	NA	NA	941.176 Кбит/с	1.0625	2.12%
10	1.792 Мбит/с	NA	NA	NA	NA	NA	NA
11	1.8432 Мбит/с	NA	NA	NA	NA	NA	NA
12	3.584 Мбит/с	NA	NA	NA	NA	NA	NA
13	3.6864 Мбит/с	NA	NA	NA	NA	NA	NA
14	7.168 Мбит/с	NA	NA	NA	NA	NA	NA
15	7.3728 Мбит/с	NA	NA	NA	NA	NA	NA

1. Чем ниже тактовая частота центрального процессора, тем меньше точность для конкретного значения скорости передачи. Верхний предел достижимой скорости передачи может быть зафиксирован с этими данными.

**Таблица. Определение погрешности для запрограммированных скоростей передачи при $f_{PCLK} = 8$ МГц
или $f_{PCLK} = 16$ МГц,
передискретизация на 8(1)**

Передискретизация на 8 (OVER8=0)							
Скорость передачи		$f_{PCLK} = 8$ МГц			$f_{PCLK} = 16$ МГц		
№	Требуемая	Реальная	Значение, запрограммированное в регистре скорости передачи	% погрешности = (Вычисленное значение – Требуемое значение) x скорость передачи / желаемую скорость передачи	Реальная	Значение, запрограммированное в регистре скорости передачи	% погрешности
1	2.4 Кбит/с	2.400 Кбит/с	416.625	0.01%	2.400 Кбит/с	833.375	0.00%
2	9.6 Кбит/с	9.604 Кбит/с	104.125	0.04%	9.598 Кбит/с	208.375	0.02%
3	19.2 Кбит/с	19.185 Кбит/с	52.125	0.08%	19.208 Кбит/с	104.125	0.04%

**Таблица. Определение погрешности для запрограммированных скоростей передачи при $f_{PCLK} = 8$ МГц
или $f_{PCLK} = 16$ МГц,
передискретизация на 8(1) (продолжение)**

Передискретизация на 8 (OVER8=0)							
Скорость передачи		$f_{PCLK} = 8$ МГц			$f_{PCLK} = 16$ МГц		
№	Требуемая	Реальная	Значение, запрограммированное в регистре скорости передачи	% погрешности = (Вычисленное значение – Требуемое значение) x скорость передачи / желаемую скорость передачи	Реальная	Значение, запрограммированное в регистре скорости передачи	% погрешности
4	57.6 Кбит/с	57.557 Кбит/с	17.375	0.08%	57.554 Кбит/с	34.750	0.08%
5	115.2 Кбит/с	115.942 Кбит/с	8.625	0.64%	115.108 Кбит/с	17.375	0.08%
6	230.4 Кбит/с	228.571 Кбит/с	4.375	0.79%	231.884 Кбит/с	8.625	0.64%
7	460.8 Кбит/с	470.588 Кбит/с	2.125	2.12%	457.143 Кбит/с	4.375	0.79%
8	896 Кбит/с	888.889 Кбит/с	1.125	0.79%	888.889 Кбит/с	2.250	0.79%
9	921.6 Кбит/с	888.889 Кбит/с	1.125	3.55%	941.176 Кбит/с	2.125	2.12%
10	1.792 Мбит/с	NA	NA	NA	1.7777 Мбит/с	1.125	0.79%
11	1.8432 Мбит/с	NA	NA	NA	1.7777 Мбит/с	1.125	3.55%
12	3.584 Мбит/с	NA	NA	NA	NA	NA	NA
13	3.6864 Мбит/с	NA	NA	NA	NA	NA	NA
14	7.168 Мбит/с	NA	NA	NA	NA	NA	NA
15	7.3728 Мбит/с	NA	NA	NA	NA	NA	NA

1. Чем ниже тактовая частота центрального процессора, тем меньше точность для конкретного значения скорости передачи. Верхний предел достижимой скорости передачи может быть зафиксирован с этими данными.

**Таблица. Определение погрешности для запрограммированных скоростей передачи при $f_{PCLK} = 30$ МГц
или $f_{PCLK} = 60$ МГц,**

передискретизация на 16(1)(2)

Передискретизация на 16 (OVER8=0)							
Скорость передачи		f _{PCLK} = 30 МГц			f _{PCLK} = 60 МГц		
№	Требуемая	Реальная	Значение, запрограммированное в регистре скорости передачи	% погрешности = (Вычисленное значение – Требуемое значение) x скорость передачи / желаемую скорость передачи	Реальная	Значение, запрограммированное в регистре скорости передачи	% погрешности
1	2.4 Кбит/с	2.400 Кбит/с	781.2500	0.00%	2.400 Кбит/с	1562.5000	0.00%
2	9.6 Кбит/с	9.600 Кбит/с	195.3125	0.00%	9.600 Кбит/с	390.6250	0.00%
3	19.2 Кбит/с	19.194 Кбит/с	97.6875	0.03%	19.200 Кбит/с	195.3125	0.00%
4	57.6 Кбит/с	57.582 Кбит/с	32.5625	0.03%	57.582 Кбит/с	65.1250	0.03%
5	115.2 Кбит/с	115.385 Кбит/с	16.2500	0.16%	115.163 Кбит/с	32.5625	0.03%
6	230.4 Кбит/с	230.769 Кбит/с	8.1250	0.16%	230.769 Кбит/с	16.2500	0.16%
7	460.8 Кбит/с	461.538 Кбит/с	4.0625	0.16%	461.538 Кбит/с	8.1250	0.16%
8	896 Кбит/с	909.091 Кбит/с	2.0625	1.46%	895.522 Кбит/с	4.1875	0.05%

Таблица. Определение погрешности для запрограммированных скоростей передачи при f_{PCLK} = 30 МГц или f_{PCLK} = 60 МГц,

передискретизация на 16(1)(2) (продолжение)

Передискретизация на 16 (OVER8=0)							
Скорость передачи		f _{PCLK} = 30 МГц			f _{PCLK} = 60 МГц		
№	Требуемая	Реальная	Значение, запрограммированное в регистре скорости передачи	% погрешности = (Вычисленное значение – Требуемое значение) x скорость передачи / желаемую скорость передачи	Реальная	Значение, запрограммированное в регистре скорости передачи	% погрешности
9	921.6 Кбит/с	909.091 Кбит/с	2.0625	1.36%	923.077 Кбит/с	4.0625	0.16%
10	1.792 Мбит/с	1.1764 Мбит/с	1.0625	1.52%	1.8182 Мбит/с	2.0625	1.36%
11	1.8432 Мбит/с	1.8750 Мбит/с	1.0000	1.73%	1.8182 Мбит/с	2.0625	1.52%
12	3.584 Мбит/с	NA	NA	NA	3.2594 Мбит/с	1.0625	1.52%
13	3.6864 Мбит/с	NA	NA	NA	3.7500 Мбит/с	1.0000	1.73%
14	7.168 Мбит/с	NA	NA	NA	NA	NA	NA
15	7.3728 Мбит/с	NA	NA	NA	NA	NA	NA

1. Чем ниже тактовая частота центрального процессора, тем меньше точность для конкретного значения скорости передачи. Верхний предел достижимой скорости передачи может быть зафиксирован с этими данными.

Таблица. Определение погрешности для запрограммированных скоростей передачи при f_{PCLK} = 30 МГц или f_{PCLK} = 60 МГц,

передискретизация на 8(1) (2)

Передискретизация на 8 (OVER8=0)							
Скорость передачи		f _{PCLK} = 30 МГц			f _{PCLK} = 60 МГц		
№	Требуемая	Реальная	Значение, запрограммированное в регистре скорости передачи	% погрешности = (Вычисленное значение – Требуемое значение) x скорость передачи / желаемую скорость передачи	Реальная	Значение, запрограммированное в регистре скорости передачи	% погрешности
1	2.4 Кбит/с	2.400 Кбит/с	1562.5000	0.00%	2.400 Кбит/с	3125.0000	0.00%
2	9.6 Кбит/с	9.600 Кбит/с	390.6250	0.00%	9.600 Кбит/с	781.2500	0.00%
3	19.2 Кбит/с	19.194 Кбит/с	195.3750	0.03%	19.200 Кбит/с	390.6250	0.00%
4	57.6 Кбит/с	57.582 Кбит/с	65.1250	0.16%	57.582 Кбит/с	130.2500	0.03%
5	115.2 Кбит/с	115.385 Кбит/с	32.5000	0.16%	115.163 Кбит/с	65.1250	0.03%
6	230.4 Кбит/с	230.769 Кбит/с	16.2500	0.16%	230.769 Кбит/с	32.5000	0.16%
7	460.8 Кбит/с	461.538 Кбит/с	8.1250	0.16%	461.538 Кбит/с	16.2500	0.16%
8	896 Кбит/с	909.091 Кбит/с	4.1250	1.46%	895.522 Кбит/с	8.3750	0.05%
9	921.6 Кбит/с	909.091 Кбит/с	4.1250	1.36%	923.077 Кбит/с	8.1250	0.16%
10	1.792 Мбит/с	1.7647 Мбит/с	2.1250	1.52%	1.8182 Мбит/с	4.1250	1.46%

**Таблица. Определение погрешности для запрограммированных скоростей передачи при $f_{PCLK} = 30$ МГц
или $f_{PCLK} = 60$ МГц,
передискретизация на 8(1) (2) (продолжение)**

Передискретизация на 8 (OVER8=0)							
Скорость передачи		$f_{PCLK} = 30$ МГц			$f_{PCLK} = 60$ МГц		
№	Требуемая	Реальная	Значение, запрограммированное в регистре скорости передачи	% погрешности = (Вычисленное значение – Требуемое значение) x скорость передачи / желаемую скорость передачи	Реальная	Значение, запрограммированное в регистре скорости передачи	% погрешности
11	1.8432 Мбит/с	1.8750 Мбит/с	2.0000	1.73%	1.8182 Мбит/с	4.1250	1.36%
12	3.584 Мбит/с	3.7500 Мбит/с	1.0000	4.63%	3.5294 Мбит/с	2.1250	1.52%
13	3.6864 Мбит/с	3.7500 Мбит/с	1.0000	1.73%	3.7500 Мбит/с	2.0000	1.73%
14	7.168 Мбит/с	NA	NA	NA	7.5000 Мбит/с	1.0000	4.63%
15	7.3728 Мбит/с	NA	NA	NA	7.5000 Мбит/с	1.0000	1.73%

1. Чем ниже тактовая частота центрального процессора, тем меньше точность для конкретного значения скорости передачи. Верхний предел достижимой скорости передачи может быть зафиксирован с этими данными.

**Таблица. Определение погрешности для запрограммированных скоростей передачи при $f_{PCLK} = 42$ МГц
или $f_{PCLK} = 84$ Гц,
передискретизация на 16(1)(2)**

Передискретизация на 16 (OVER8=0)							
Скорость передачи		$f_{PCLK} = 42$ МГц			$f_{PCLK} = 84$ МГц		
№	Требуемая	Реальная	Значение, запрограммированное в регистре скорости передачи	% погрешности = (Вычисленное значение – Требуемое значение) x скорость передачи / желаемую скорость передачи	Реальная	Значение, запрограммированное в регистре скорости передачи	% погрешности
1	1.2 Кбит/с	1.2 Кбит/с	2187.5	0	1.2 Кбит/с	NA	0
2	2.4 Кбит/с	2.4 Кбит/с	1093.75	0	2.4 Кбит/с	2187.5	0
3	9.6 Кбит/с	9.6 Кбит/с	273.4375	0	9.6 Кбит/с	546.875	0
4	19.2 Кбит/с	19.195 Кбит/с	136.75	0.02	19.2 Кбит/с	273.4375	0
5	38.4 Кбит/с	38.391 Кбит/с	68.375	0.02	38.391 Кбит/с	136.75	0.02
6	57.6 Кбит/с	57.613 Кбит/с	45.5625	0.02	57.613 Кбит/с	91.125	0.02
7	115.2 Кбит/с	115.068 Кбит/с	22.8125	0.11	115.226 Кбит/с	45.5625	0.02
8	230.4 Кбит/с	230.769 Кбит/с	11.375	0.16	230.137 Кбит/с	22.8125	0.11
9	460.8 Кбит/с	461.538 Кбит/с	5.6875	0.16	461.538 Кбит/с	11.375	0.16
10	921.6 Кбит/с	913.043 Кбит/с	2.875	0.93	923.076 Кбит/с	5.6875	0.93
11	1.792 Мбит/с	1.826 Мбит/с	1.4375	1.9	1.787 Мбит/с	2.9375	0.27
12	1.8432 Мбит/с	1.826 Мбит/с	1.4375	0.93	1.826 Мбит/с	2.875	0.93
13	3.584 Мбит/с	NA	NA	NA	3.652 Мбит/с	1.4375	1.9
14	3.6864 Мбит/с	NA	NA	NA	3.652 Мбит/с	1.4375	0.93

**Таблица. Определение погрешности для запрограммированных скоростей передачи при $f_{PCLK} = 42$ МГц
или $f_{PCLK} = 84$ Гц,
передискретизация на 16(1)(2) (продолжение)**

Передискретизация на 16 (OVER8=0)							
Скорость передачи		$f_{PCLK} = 42$ МГц			$f_{PCLK} = 84$ МГц		
№	Требуемая	Реальная	Значение, запрограммированное в регистре скорости передачи	% погрешности = (Вычисленное значение – Требуемое значение) x скорость передачи / желаемую скорость передачи	Реальная	Значение, запрограммированное в регистре скорости передачи	% погрешности
15	7.168 Мбит/с	NA	NA	NA	NA	NA	NA
16	7.3728 Мбит/с	NA	NA	NA	NA	NA	NA
17	9 Мбит/с	NA	NA	NA	NA	NA	NA
18	10.5 Мбит/с	NA	NA	NA	NA	NA	NA

1. Чем ниже тактовая частота центрального процессора, тем меньше точность для конкретного значения скорости передачи. Верхний предел достижимой скорости передачи может быть зафиксирован с этими данными.

**Таблица. Определение погрешности для запрограммированных скоростей передачи при $f_{PCLK} = 42$ МГц
или $f_{PCLK} = 84$ МГц,
передискретизация на 8(1)(2)**

Передискретизация на 8 (OVER8=0)							
Скорость передачи		$f_{PCLK} = 42$ МГц			$f_{PCLK} = 84$ МГц		
№	Требуемая	Реальная	Значение, запрограммированное в регистре скорости передачи	% погрешности = (Вычисленное значение – Требуемое значение) x скорость передачи / желаемую скорость передачи	Реальная	Значение, запрограммированное в регистре скорости передачи	% погрешности
1	2.4 Кбит/с	2.4 Кбит/с	2187.5	0	2.4 Кбит/с	NA	0
2	9.6 Кбит/с	9.6 Кбит/с	546.875	0	9.6 Кбит/с	1093.75	0
3	19.2 Кбит/с	19.195 Кбит/с	273.5	0.02	19.2 Кбит/с	546.875	0
4	38.4 Кбит/с	38.391 Кбит/с	136.75	0.02	38.391 Кбит/с	273.5	0.02
5	57.6 Кбит/с	57.613 Кбит/с	91.125	0.02	57.613 Кбит/с	182.25	0.02
6	115.2 Кбит/с	115.068 Кбит/с	45.625	0.11	115.226 Кбит/с	91.125	0.02
7	230.4 Кбит/с	230.769 Кбит/с	22.75	0.11	230.137 Кбит/с	45.625	0.11
8	460.8 Кбит/с	461.538 Кбит/с	11.375	0.16	461.538 Кбит/с	22.75	0.16
9	921.6 Кбит/с	913.043 Кбит/с	5.75	0.93	923.076 Кбит/с	11.375	0.93
10	1.792 Мбит/с	1.826 Мбит/с	2.875	1.9	1.787 Мбит/с	5.875	0.27
11	1.8432 Мбит/с	1.826 Мбит/с	2.875	0.93	1.826 Мбит/с	5.75	0.93
12	3.584 Мбит/с	3.5 Мбит/с	1.5	2.34	3.652 Мбит/с	2.875	1.9
13	3.6864 Мбит/с	3.82 Мбит/с	1.375	3.57	3.652 Мбит/с	2.875	0.93
14	7.168 Мбит/с	NA	NA	NA	7 Мбит/с	1.5	2.34
15	7.3728 Мбит/с	NA	NA	NA	7.636 Мбит/с	1.375	3.57

**Таблица. Определение погрешности для запрограммированных скоростей передачи при $f_{PCLK} = 42$ МГц
или $f_{PCLK} = 84$ МГц,
передискретизация на 8(1)(2) (продолжение)**

Передискретизация на 8 (OVER8=0)							
Скорость передачи		$f_{PCLK} = 42$ МГц			$f_{PCLK} = 84$ МГц		
№	Требуемая	Реальная	Значение, запрограммированное в регистре скорости передачи	% погрешности = (Вычисленное значение – Требуемое значение) x скорость передачи / желаемую скорость передачи	Реальная	Значение, запрограммированное в регистре скорости передачи	% погрешности
16	9 Мбит/с	NA	NA	NA	9.333 Мбит/с	1.125	3.7
17	10.5 Мбит/с	NA	NA	NA	10.5 Мбит/с	1	0

1. Чем ниже тактовая частота центрального процессора, тем меньше точность для конкретного значения скорости передачи. Верхний предел достижимой скорости передачи может быть зафиксирован с этими данными.

Допуск ухода тактовой частоты для приемника USART

Асинхронный приемник USART корректно работает, только если общее отклонение тактовой частоты меньше, чем допуск приемника USART. Причины, которые способствуют общему отклонению:

- DTRA: отклонение из-за ошибки передатчика (что также включает отклонение от локального тактового генератора передатчика)
- DQUANT: ошибка квантования скорости приемника
- DREC: отклонение локального генератора тактов приемника
- DTCL: отклонение из-за линии передачи (обычно из-за того, что приемопередатчики могут давать ассиметричные перепады от 0 к 1 по сравнению с перепадами от 1 к 0)

$$DTRA + DQUANT + DREC + DTCL < \text{допуск приемника USART}$$

Допуск приемника USART для правильного приема данных равен максимально допустимому отклонению и зависит от следующих параметров:

- Длина символа 10- или 11-бит, что определяется битом М в регистре USART_CR1
- Передискретизация на 8 или 16, что определяется битом OVER8 в регистре USART_CR1
- Используется или нет дробная установка скорости
- Используется 1 или 3 бита для оцифровки данных, в зависимости от бита ONEBIT в регистре USART_CR3

Таблица. Допуск приемника USART при DIV_fraction = 0

Бит М	OVER8 = 0		OVER8 = 1	
	ONEBIT=0	ONEBIT=1	ONEBIT=0	ONEBIT=1
0	3.75%	4.375%	2.50%	3.75%
1	3.41%	3.97%	2.27%	3.41%

Таблица. Допуск приемника USART при DIV_Fraction, не равным 0

Бит М	OVER8 = 0		OVER8 = 1	
	ONEBIT=0	ONEBIT=1	ONEBIT=0	ONEBIT=1
0	3.33%	3.88%	2%	3%
1	3.03%	3.53%	1.82%	2.73%

Многопроцессорный обмен

Существует возможность многопроцессорного обмена данными через USART (несколько приемопередатчиков USART, объединенных в одну сеть). Например, один USART может быть ведущим устройством, его выход TX подключается ко входу RX другого приемопередатчика USART. Другие приемопередатчики являются ведомыми, их соответствующие выходы TX логически объединены операцией «логическое И» и подсоединены ко входу RX ведущего устройства.

В многопроцессорных конфигурациях предпочтительно, чтобы только один получатель сообщения активно принимал все сообщение целиком, чтобы уменьшить обработку избыточных данных для приемников, которым эти данные не адресованы.

Неадресованные устройства могут быть переведены в режим «молчания» (mute mode) при помощи функции приостановки. В данном режиме:

- Никакой статус бит приема не может быть установлен.
- Все прерывания приема запрещены.
- Бит RWU в регистре USART_CR1 установлен в 1. Бит RWU может управляться автоматически аппаратно или может быть записан программой при определенных условиях.

USART может войти в режим молчания или выйти из него при помощи одного из двух методов, в зависимости от значения бита WAKE в регистре USART_CR1:

- Обнаружение линии Idle, если бит WAKE сброшен,
- Обнаружение сигнала метки адреса (Address Mark), если бит WAKE установлен.

Обнаружение линии Idle (WAKE=0)

USART переключится в режим молчания, если бит RWU установлен в 1.

Приемопередатчик выйдет из режима молчания, когда будет обнаружен фрейм Idle. После чего бит RWU сбрасывается аппаратной частью, но бит IDLE в регистре USART_SR не устанавливается. Бит RWU может быть записан в 0 программно.

На Рисунке 116 показан пример поведения устройства в режиме молчания с использованием обнаружения линии Idle.

Рисунок 116. Режим молчания (Mute) с обнаружением линии Idle



Обнаружение метки адреса (Address mark) (WAKE=1)

В данном режиме, байты распознаются как адреса, если СЗБ равен 1, в другом случае они распознаются как данные. В байте адреса адрес приемника помещается в 4 МЗБ. Приемник сравнивает это 4-битное слово с собственным адресом, запрограммированным в битах ADD в регистре USART_CR2.

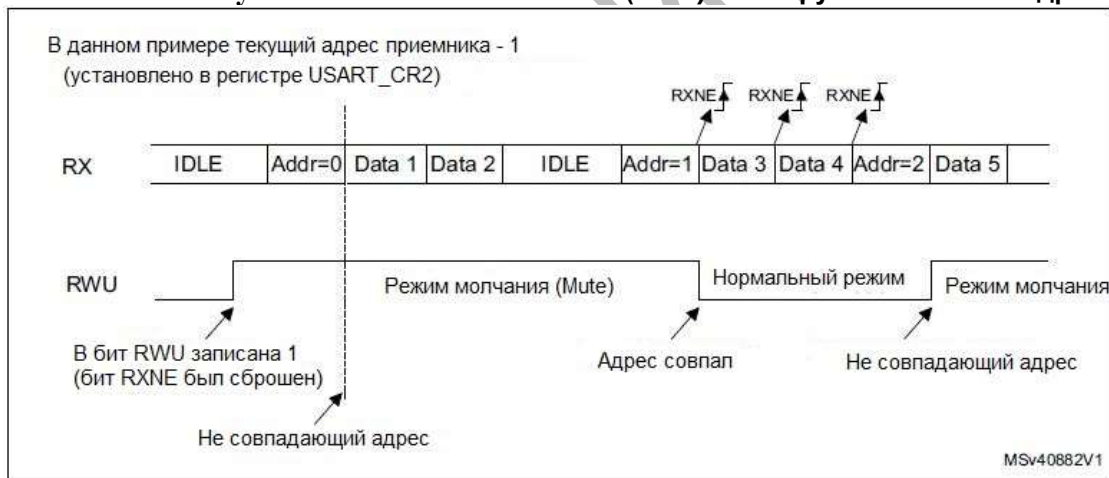
USART переключится в режим молчания, когда будет получен символ адреса, который не соответствует запрограммированному режиму. В данном случае, бит RWU устанавливается аппаратно. Флаг RXNE не установится для данного байта адреса, и не будет сгенерировано прерывание, так как приемопередатчик перейдет в режим молчания.

Приемопередатчик выйдет из режима молчания, когда будет принят символ адреса, совпадающий с запрограммированным адресом. После чего бит RWU будет сброшен и последующие биты будут приняты. Символ адреса будет установлен в бите RXNE, так как бит RWU был сброшен.

Бит RWU может быть записан в 0 или 1, когда буфер приемника не содержит данных (RXNE=0 в регистре USART_SR). Иначе попытка записи будет проигнорирована.

Пример поведения приемопередатчика в режиме молчания с использованием режима обнаружения метки адреса приведен на Рисунке 117.

Рисунок 117. Режим молчания (Mute) с обнаружением метки адреса



Контроль четности

Контроль четности бит (генерация бита четности при передаче и контроль бита четности при приеме) разрешается установкой бита PCE в регистре USART_CR1. В зависимости от длины фрейма, определяемой битом M, возможные форматы фрейма приемопередатчика USART приведены в таблице ниже.

Таблица 146. Форматы фрейма

Бит M	Бит PCE	Фрейм USART (1)
0	0	SB 8 бит данных STB
0	1	SB 7 бит данных PB STB
1	0	SB 9 бит данных STB
1	1	SB 8 бит данных PB STB

1. SB: стартовый бит, STB: стоп-бит, PB: бит четности.

Проверка на четность

Бит четности вычисляется так, чтобы получить четную сумму всех «1» во фрейме из 7 или 8 бит (в зависимости от значения бита M) и бита четности.

Например, data=00110101; установлено 4 бита => бит четности = 0, если выбрана проверка на четность (бит PS в USART_CR1 = 0).

Проверка на нечетность

Бит четности вычисляется так, чтобы получить нечетную сумму всех «1» во фрейме из 7 или 8 бит (в зависимости от значения бита M) и бита четности.

Например, data=00110101; установлено 4 бита => бит четности = 1, если выбрана проверка на нечетность (бит PS в USART_CR1 = 1).

Проверка четности при приеме

Если проверка четности показала ошибку данных, устанавливается флаг PE в регистре USART_SR, и генерируется прерывание, если в регистре USART_CR1 установлен бит PEIE. Флаг PE сбрасывается программной последовательностью (чтение регистра статуса, за которым следует чтение или запись регистра данных USART_DR).

Примечание: в случае пробуждения по метке адреса: MSB используется для идентификации адреса, а не бит четности. Приемник не проверяет бит четности данных адреса (бит PE не устанавливается в случае ошибки четности).

Генерация бита четности при передаче

Если в регистре USART_CR1 установлен бит PCE, то СЗБ данных, записанных в регистре данных, будет передан, но он будет изменен битом четности (четное количество «1», если

выбрана проверка на четность ($PS=0$) или нечетное количество «1», если выбрана проверка на нечетность ($PS=1$)).

Примечание: Программная часть, которая контролирует передачу, может активировать программную последовательность для сброса флага PE (чтение регистра статуса, за которым следует доступ на чтение или запись регистра данных). При работе в полудуплексном режиме, в зависимости от программной части, это может привести к неожиданному сбросу флага PE.

ПРЕДВАРИТЕЛЬНО

Синхронный режим USART

Синхронный режим выбирается записью бита CLKEN в «1» в регистре USART_CR2.

В синхронном режиме следующие биты должны быть сброшены:

- SCEN и HDSEL в регистре USART_CR3.

USART дает возможность пользователю управлять двунаправленным синхронным последовательным обменом данных в режиме ведущего приемопередатчика. Вывод СК является выходом тактов передатчика USART. Во время стартового и стоп-битов никакие тактовые импульсы не посылаются на вывод СК. В зависимости от состояния бита LBCL в регистре USART_CR2 тактовые импульсы будут или не будут генерироваться во время последнего достоверного бита данных (маркер адреса). Бит CPOL в регистре USART_CR2 позволяет пользователю выбрать полярность тактов, а бит CPHA в регистре USART_CR2 позволяет выбрать фазу внешних тактов (см. Рисунок 118, Рисунок 119 и Рисунок 120).

Во время состояния ожидания (Idle), преамбулы и отправки символа break, внешний тактовый сигнал СК не активируется.

В синхронном режиме, передатчик работает точно также, как и в асинхронном режиме. Но так как сигнал СК синхронизируется с сигналом TX (в зависимости от CPOL и CPHA), данные на TX синхронные.

Приемник в данном режиме работает не так, как в асинхронном режиме. Если RE=1, данные тактируются синхронно с сигналом СК (передний и задний фронт, в зависимости от значений CPOL и CPHA), без какой-либо передискретизации. Время установки и удержания сигнала должно соблюдаться (зависит от скорости обмена: 1/16 от времени бита).

Примечание: Вывод СК работает совместно с выводом TX. Таким образом, тактирование обеспечивается только, если разрешен передатчик (TE=1) и передаются данные (регистр данных USART_DR записан). Это означает, что невозможно принимать синхронные данные без передачи данных.

Должны быть выбраны биты LBCL, CPOL и CPHA, когда передатчик и приемник отключены (TE=RE=0) для обеспечения корректной работы тактовых импульсов. Значения данных бит не должны изменяться во время включения передатчика или приемника.

Рекомендуется устанавливать биты TE и RE одной командой с целью минимизации времени установки и удержания приемника.

Приемопередатчик USART поддерживает только режим ведущего устройства: он не может принимать или передавать данные, связанные со входом тактирования (СК всегда работает как выход).

Рисунок 118. Пример синхронной передачи USART



Рисунок 119. Диаграмма тактирования данных USART (M=0)

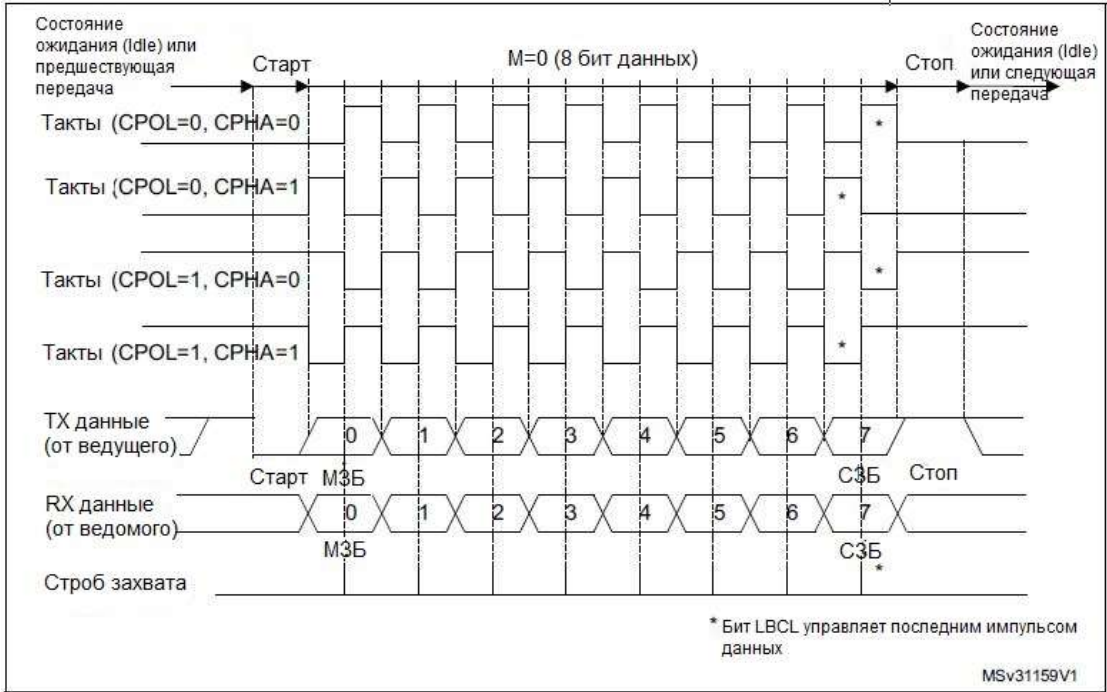


Рисунок 120. Диаграмма тактирования данных USART (M=1)

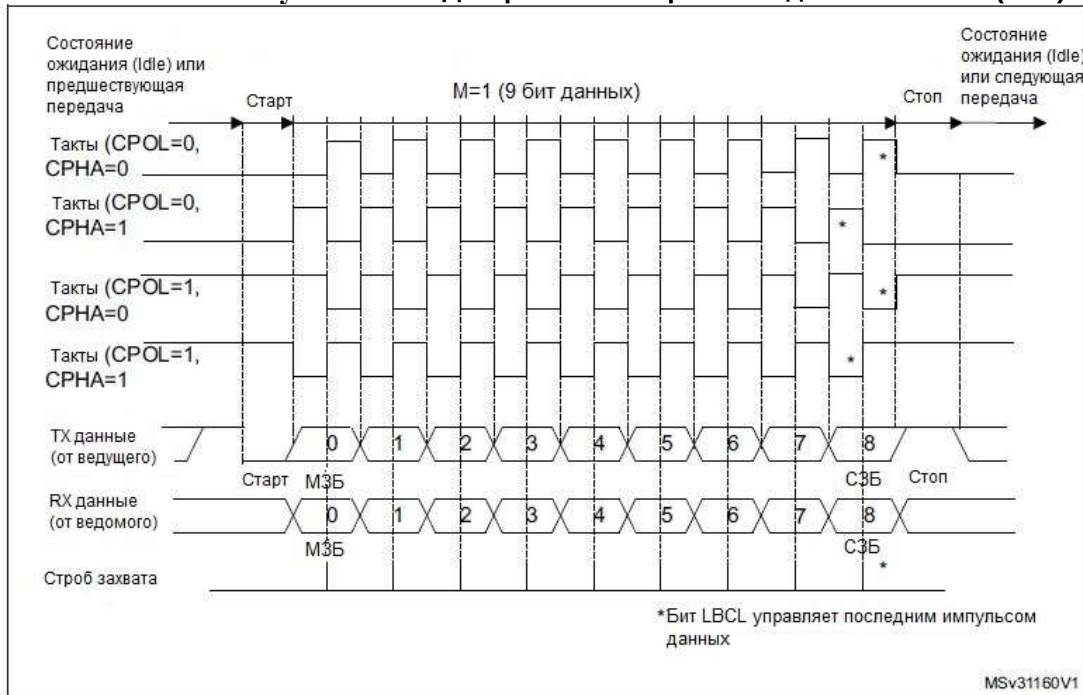
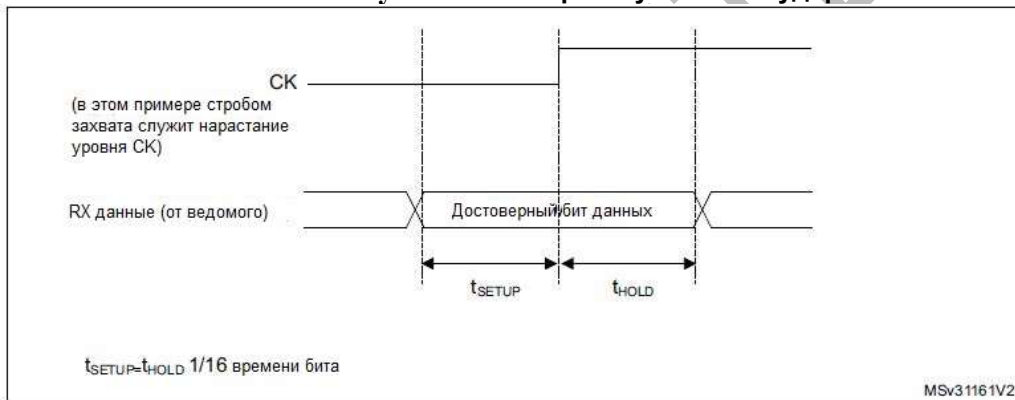


Рисунок 121. Время установки/удержания RX



Примечание: Функция тактирования СК отличается в режиме Smartcard (См. далее описание режима Smartcard).

Однопроводной полудуплексный обмен данными

Однопроводной полудуплексный режим обмена данными (single-wire half-duplex mode) выбирается путем установки бита HDSEL в регистре USART_CR3. В данном режиме должны быть сброшены следующие биты:

- SCEN в регистре USART_CR3.

USART может быть сконфигурирован для соответствия однопроводному полудуплексному протоколу, где линии сигналов TX и RX имеют внутреннее соединение. Выбор между полудуплексным или полнодуплексным режимом передачи данных осуществляется управляющим битом 'HALF DUPLEX SEL' (HDSEL в регистре USART_CR3).

Как только бит HDSEL устанавливается в «1»:

- линии сигналов TX и RX соединяются внутри
- вывод RX больше не используется
- вывод TX всегда свободен, если данные не передаются. Таким образом, он используется как стандартный ввод/вывод в режиме ожидания или при приеме данных. Это означает, что данный ввод/вывод должен быть сконфигурирован так, чтобы он был высокоомным входом (или выходом с открытым стоком), когда он не управляется USART.

Кроме этого, обмен данными происходит так же, как и обычном режиме передачи данных USART.

Конфликты на линии связи должны контролироваться программно (например, с использованием централизованного арбитра шины). В частности, передача данных никогда не блокируется аппаратной частью и будет продолжаться, как только данные будут записаны в регистр данных, пока установлен бит TE.

Режим Smartcard

Режим смарт-карты (Smartcard mode) выбирается путем установки бита SCEN в регистре USART_CR3. В режиме Smartcard следующие биты должны быть сброшены:

- бит HDSEL в регистре USART_CR3.

Более того, бит CLKEN может быть установлен с целью тактирования смарт-карты.

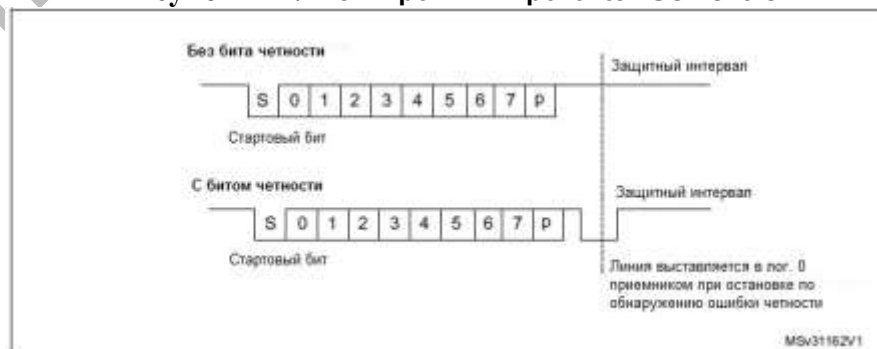
Интерфейс режима Smartcard разработан для поддержки асинхронного протокола в соответствии со стандартом ISO 7816-3. USART должен быть сконфигурирован следующим образом:

- 8 бит + бит четности: где M=1 и PCE=1 в регистре USART_CR1
- 1.5 стоп бита при передаче и приеме: где STOP=11 в регистре USART_CR2.

Примечание: Также можно выбрать 0.5 стоп бит для приема, но рекомендуется использовать 1.5 стоп бита для передачи и приема данных, чтобы избежать переключения между двумя конфигурациями.

Рисунок 122 показывает пример, как выглядит сигнал данных с ошибкой четности и без ошибки четности.

Рисунок 122. Асинхронный протокол ISO 7816-3



При подключении к смарт-карте, выход TX приемопередатчика USART управляет двунаправленной линией, которая также управляется и смарт-картой. Вывод TX должен быть сконфигурирован как открытый сток.

Smartcard – это однопроводный полудуплексный протокол передачи данных.

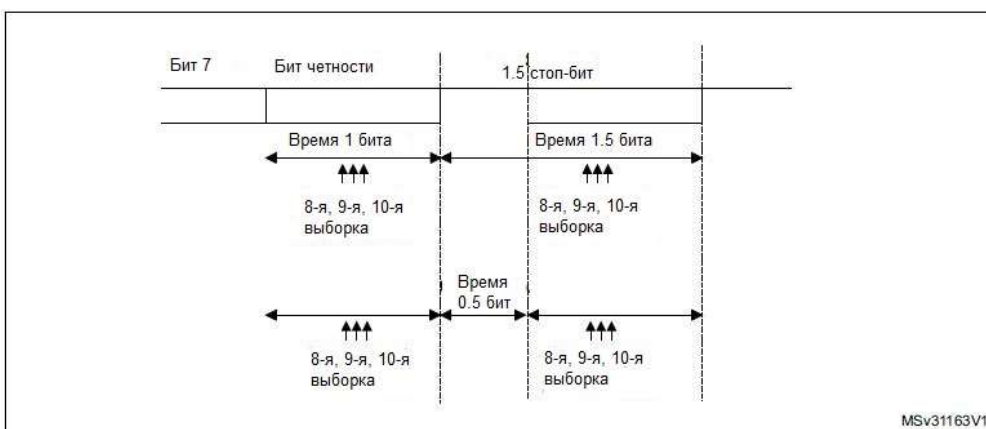
- Передача данных из регистра сдвига передатчика осуществляется с гарантированной задержкой минимум $\frac{1}{2}$ такта скорости. При нормальном режиме работы полная передача из регистра сдвига начнется сдвигом на следующем перепаде тактов скорости. В режиме Smartcard эта передача дополнительно задерживается на $\frac{1}{2}$ такта скорости.
- Если обнаружена ошибка четности при приеме фрейма, запрограммированного на 0.5 или 1.5 стоп-бита, линия передачи подтягивается в 0 на период такта скорости после завершения приема фрейма. Это показывает смарт-карте, что данные, переданные на приемопередатчик USART, не были корректно приняты. Данный сигнал NACK (подтягивание линии передачи в 0 на период 1 такта скорости) сгенерирует ошибку фрейма на стороне передатчика (конфигурация 1.5 стоп бита). Приложение должно обработать эту ситуацию повторной отправкой данных в соответствии с протоколом. Ошибка четности не подтверждается приемником (сигнал NACK), если установлен управляющий бит NACK, иначе сигнал NACK не передается.
- Установка флага TC может быть задержана путем программирования регистра защитного интервала времени (Guard Time). При нормальной работе флаг TC устанавливается, когда регистр сдвига передатчика пуст и нет никаких запросов на передачу. В режиме смарт-карты пустой регистр сдвига передатчика запускает счетчик защитного интервала для счета запрограммированного значения в регистре защитного интервала времени (Guard Time register). Флаг TC подтягивается в 0. Когда счетчик достигает запрограммированного значения, устанавливается флаг TC.
- На снятие флага TC режим смарт-карты не влияет.
- Если обнаружена ошибка фрейма на стороне передатчика (ответ NACK от приемника), сигнал NACK не будет обнаружен как стартовый бит блоком приема передатчика. В соответствии со стандартом ISO, длительность принятого сигнала NACK может составлять 1 или 2 периода тактов скорости.
- На стороне приемника, если была обнаружена ошибка четности, и был передан сигнал NACK, приемник не обработает сигнал NACK как стартовый бит.

Примечание: Символ break не имеет значения в режиме Smartcard. Данные 0x00 с ошибкой фрейма будут обработаны как данные, а не как символ break.

Фрейм Idle не передается при переключении бита TE. Фрейм Idle (как определено для других конфигураций) не определен в протоколе ISO.

Рисунок 123 показывает, как обрабатывается сигнал NACK. В данном примере USART передает данные и сконфигурирован на 1.5 стоп бита. Блок приемника USART разрешен для проверки целостности данных и сигнала NACK.

Рисунок 123. Обнаружение ошибки четности с использованием 1.5 стоп-бит



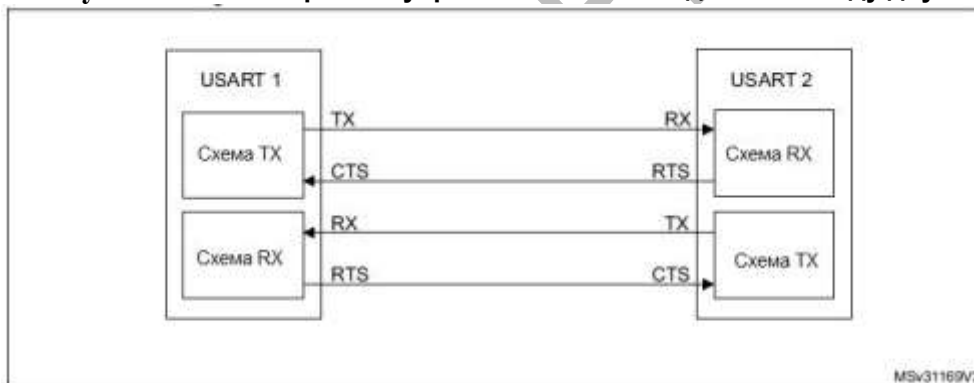
USART может тактировать смарт-карту при помощи выхода СК. В режиме смарт-карты, выход

СК не связан с обменом данными, он просто генерирует тактовую частоту, полученную из внутреннего входа тактирования периферии при помощи 5-битного делителя. Коэффициент деления конфигурируется в регистре делителя USART_GTPR. Частота СК программируется от $f_{CK}/2$ до $f_{CK}/62$, где f_{CK} это частота тактов входа периферии.

Аппаратное управление потоком

Можно управлять потоком последовательных данных между двумя устройствами, используя вход CTS и выход RTS. Рисунок 124 показывает схему соединения двух устройств для данного режима:

Рисунок 124. Аппаратное управление потоком данных между двумя устройствами USART



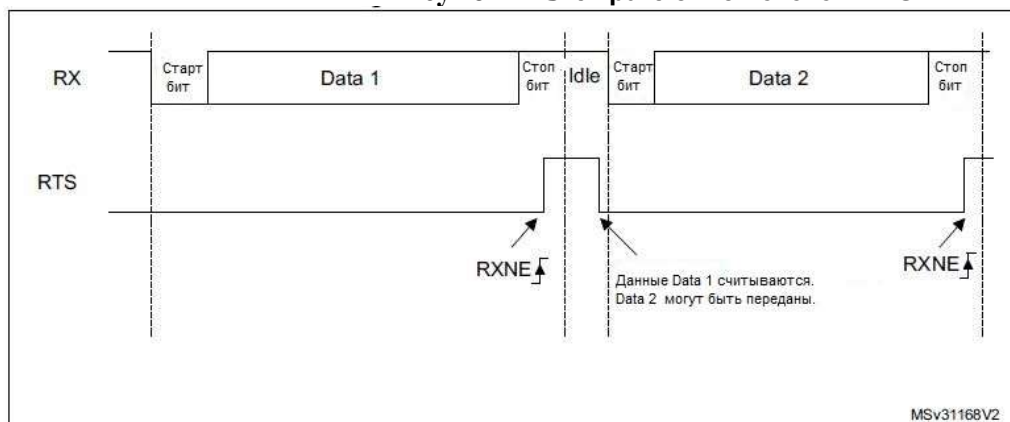
Управление потоком RTS и CTS можно разрешить независимо записью битов RTSE и CTSE соответственно в 1 (в регистре USART_CR3).

Управление потоком RTS

Если разрешено управление потоком RTS ($RTSE=1$), выставляется сигнал RTS (лог. 0), когда приемник USART готов принимать новые данные. Если регистр приема заполнен, то сигнал RTS снимается, тем самым показывая, что ожидается остановка передачи в конце текущего фрейма.

Рисунок 124 показывает пример обмена с разрешенным управлением потоком RTS.

Рисунок 125. Управление потоком RTS



Управление потоком CTS

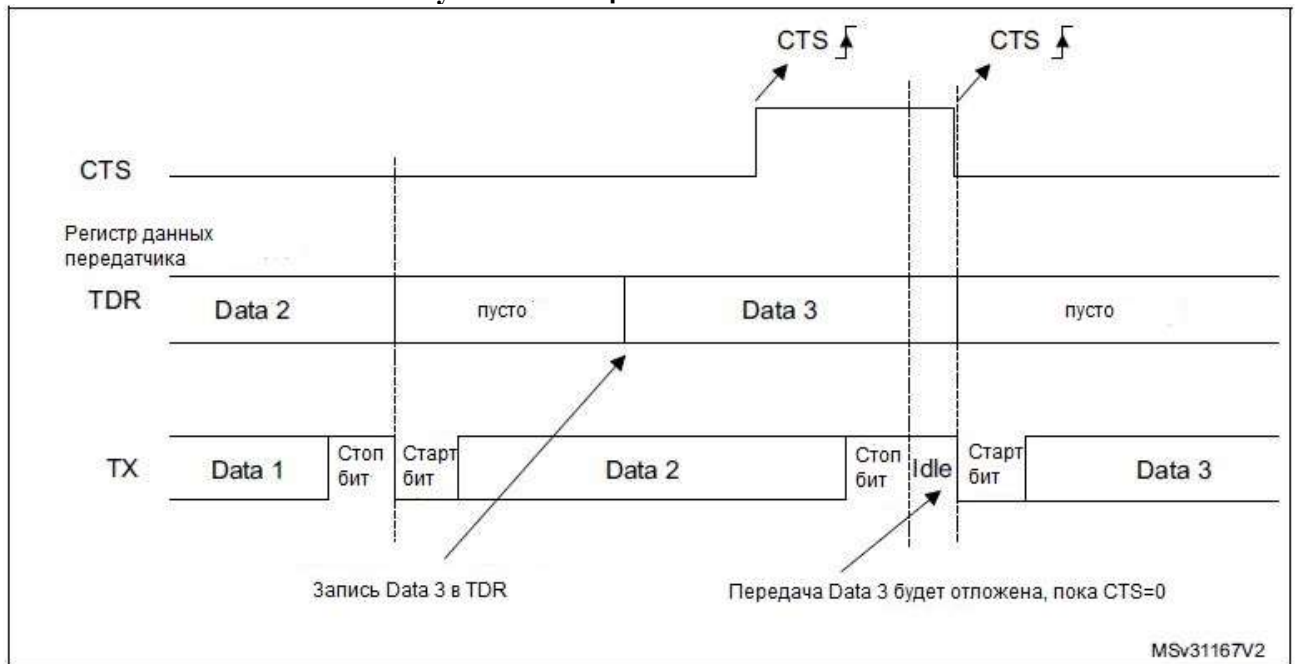
Если управление потоком CTS разрешено ($CTSE=1$), передатчик проверяет вход CTS перед передачей следующего фрейма. Если устанавливается сигнал CTS (лог. 0), тогда начинается передача следующих данных (подразумевается, что есть данные для передачи, т.е. другими словами

$TXE=0$), в ином случае передача не произойдет. Когда снимается сигнал CTS во время передачи, текущая передача завершится перед остановкой передатчика.

Если $CTSE=1$, статус CTSIF будет установлен автоматически аппаратурой, как только вход CTS

переключится. Это показывает, когда приемник будет готов к обмену данными. Прерывание генерируется, если установлен бит CTSIE в регистре USART_CR3. Рисунок ниже показывает пример обмена с разрешенным управлением потоком CTS.

Рисунок 126. Управление потоком CTS



Примечание: **Особое поведение фреймов break:** при разрешенном управлении потоком CTS, передатчик не проверяет статус входа CTS для отправки сигнала break

Прерывания USART

Table 147. Запрос на прерывание USART

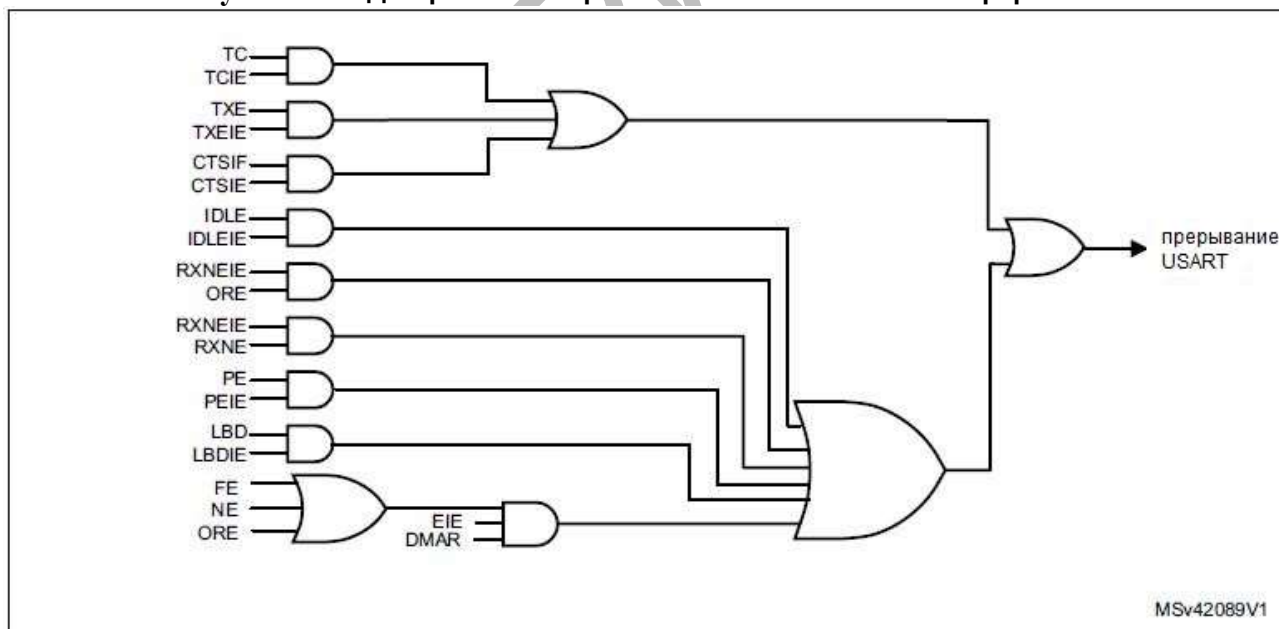
Событие	Флаг события	Бит разрешения
Регистр данных передачи пуст	TXE	TXEIE
Флаг CTS	CTS	CTSIE
Передача завершена	TC	TCIE
Принятые данные готовы для чтения	RXNE	RXNEIE
Обнаружена ошибка переполнения	ORE	
Обнаружена линия Idle	IDLE	IDLEIE
Ошибка четности	PE	PEIE
Флаг Break	LBD	LBDIE
Флаг шума, ошибки переполнения или ошибки фрейма в случае мультибуферного обмена	NF или ORE или FE	EIE

События прерываний подсоединены к одному и тому же вектору прерывания (см. Рисунок 127).

- Во время передачи: Передача завершена (Transmission Complete), Линия пуста для начала передачи (Clear to Send) или Регистр данных передачи пуст (Transmit Data Register Empty).
- Во время приема: Обнаружение линии Idle, ошибка переполнения, Регистр данных приема не пуст (Receive Data register not empty), Ошибка четности, флаг шума (только при мультибуферном обмене) и ошибка фрейма (только при мультибуферном обмене).

Все эти события генерируют прерывания, если соответствующий бит разрешения прерывания установлен.

Рисунок 127. Диаграмма отображений событий USART на прерывание



Регистры USART

Доступ к регистрам периферии осуществляется по 16-битным полусловам или 32-битными словам. Таблица ниже показывает карту памяти регистров USART и их значения после сброса.

Таблица. Регистры USART и их значения после сброса

Смещение	Регистр	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	USART_SR	Зарезервировано																						CTS	Резерв	TXE	TC	RXN	IDLE	ORE	NF	FE	PE
	0																							1									
0x04	USART_DR	Зарезервировано																						DR[8:0]									
	0																							0	0	0	0	0	0	0	0	0	0
0x08	USART_BRR	Зарезервировано												DIV_Mantissa[15:4]								DIV_Fraction[3:0]											
	0													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					
0x0C	USART_CR1	Зарезервировано												OVER8	зарезервировано	UE	M	WAKE	PCE	PS	PEIE	TXEIE	TCIE	RXNEIE	IDLEIE	TE	RE	RWU	SBK				
	0													0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
0x10	USART_CR2	Зарезервировано												зарезервировано	STOP[1:0]	CLKEN		CPOL	CPHA	LBCL	зарезервировано	зарезервировано	зарезервировано	зарезервировано	ADD[3:0]								
	0														0	0	0	0	0	0					0	0	0	0	0	0			
0x14	USART_CR3	Зарезервировано												ONEBIT		CTSIE	CTSE	RTSE	DWAT	DMAR	SCEN	NACK	HDSEL	IRLP	IREN	EIE							
	0													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
0x18	USART_GTPR	Зарезервировано												GT[7:0]								PSC[4:0]											
	0													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		

Регистр статуса (USART_SR)

Смещение адреса: 0x00

Значение после сброса: 0x0000 00C0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Зарезервировано															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Зарезервировано						CTS	Res.	TXE	TC	RXNE	IDLE	ORE	NF	FE	PE
						rc_w0	-	r	rc_w0	rc_w0	r	r	r	r	r

Биты 31:10 зарезервированы, их значение должно сохраняться в состоянии сброса (все нули).

Бит 9 CTS: Флаг CTS

Данный бит устанавливается аппаратно, когда на входе CTS меняется уровень, если установлен бит CTSE. Бит сбрасывается программно записью 0. Прерывание генерируется, если CTSIE=1 в регистре USART_CR3.

0: не было изменения статуса линии CTS

1: на линии CTS обнаружено изменение уровня

Примечание: Данный бит недоступен для UART4 и UART5.

Бит 8 Зарезервировано

Бит 7 TXE: Регистр данных передатчика пуст

Данный бит устанавливается аппаратно, когда содержимое регистра TDR было перемещено в регистр смещения. Прерывание генерируется, если TXEIE=1 в регистре USART_CR1. Бит сбрасывается записью регистра USART_DR.

0: данные не были переданы в регистр сдвига

1: данные переданы в регистр сдвига

Примечание: Данный бит используется во время передачи с использованием одного буфера.

Бит 6 TC: Передача завершена

Данный бит устанавливается аппаратно, когда передача фрейма, содержащего данные, завершена, и бит TXE установлен. Прерывание генерируется, если TCIE=1 в регистре USART_CR1. Бит сбрасывается программной последовательностью (чтение регистра USART_SR, затем запись регистра USART_DR). Бит TC может быть также сброшен записью 0. Данный сброс рекомендуется выполнять только при мультибуферном обмене.

0: Передача не завершена

1: Передача завершена

Бит 5 RXNE: Регистр данных приема не пуст

Данный бит устанавливается аппаратно, когда содержимое регистра сдвига RDR было передано в регистр

USART_DR. Прерывание генерируется, если RXNEIE=1 в регистре USART_CR1. Бит сбрасывается чтением регистра USART_DR. Флаг RXNE может быть также сброшен записью 0. Данная последовательность сброса рекомендована для мультибуферного обмена.

0: данные не получены

1: полученные данные готовы для чтения

Бит 4 IDLE: Обнаружение линии IDLE (ожидание)

Данный бит устанавливается аппаратно, когда обнаружена линия Idle. Прерывание генерируется, если

IDLEIE=1 в регистре USART_CR1. Бит сбрасывается программной последовательностью (чтение регистра USART_SR, затем чтение регистра USART_DR).

0: не обнаружена линия Idle

1: линия Idle обнаружена

Примечание: Бит IDLE не будет снова установлен, пока не установится бит RXNE (т.е. пока не будет обнаружено состояние ожидания на линии).

Бит 3 ORE: Ошибка переполнения

Данный бит устанавливается аппаратно, когда принятое в настоящий момент слово в регистре сдвига готово к передаче в регистр RDR, если RXNE=1. Прерывание генерируется, если RXNEIE=1 в регистре USART_CR1. Бит сбрасывается программной последовательностью (чтение регистра USART_SR, затем чтение регистра USART_DR).

0: нет ошибки переполнения

1: обнаружена ошибка переполнения

Примечание: Когда установлен этот бит, содержимое регистра RDR не будет потеряно, но регистр сдвига будет перезаписан. Прерывание генерируется по флагу ORE в случае мультибуферного обмена, если установлен бит EIE.

Бит 2 NF: Флаг обнаружения шума

Данный бит устанавливается аппаратно, когда в принятом фрейме обнаружен шум. Бит сбрасывается программной последовательностью (чтение регистра USART_SR, затем чтение регистра USART_DR).

0: шум не обнаружен

1: шум обнаружен

Примечание: Данный бит не генерирует прерывание, так как он появляется одновременно с битом RXNE, который сам по себе генерирует прерывание по флагу NF в случае мультибуферного обмена, если установлен бит EIE.

Примечание: Если шумов на линии нет, флаг NF может быть отключен установкой бита ONEBIT в 1 для увеличения допуска на отклонения приемопередатчика (см. Допуск ухода тактовой частоты приемника USART).

Бит 1 FE: Ошибка фрейма

Данный бит устанавливается аппаратно, когда произошла рассинхронизация, чрезмерный шум или был получен символ break. Бит сбрасывается программной последовательностью (чтение регистра USART_SR, затем чтение регистра USART_DR).

0: ошибки фрейма не обнаружено

1: обнаружена ошибка фрейма или символ break

Примечание: Данный бит не генерирует прерывание, так как он появляется одновременно с битом RXNE который сам по себе генерирует прерывание. Если слово, передаваемое в настоящий момент, генерирует и ошибку фрейма, и ошибку переполнения, оно будет передано, и установится только бит ORE.

Прерывание генерируется по флагу FE в случае мультибуферного обмена, если установлен бит EIE.

Бит 0 PE: Ошибка четности

Данный бит устанавливается аппаратно, когда в режиме приема обнаружена ошибка четности. Бит сбрасывается программной последовательностью (чтение регистра статуса, затем чтение или запись регистра данных USART_DR). Программа должна дожидаться, пока установится флаг RXNE, перед тем как сбрасывать бит PE.

Прерывание генерируется, если PEIE = 1 в регистре USART_CR1. Бит PE устанавливается одновременно со всеми остальными флагами RX (RXNE, ORE, NF, FE), поэтому данный бит может сгенерировать запрос прерывания только по окончании приёма.

0: нет ошибки четности

1: обнаружена ошибка четности

Регистр данных (USART_DR)

Смещение адреса: 0x04

Значение после сброса: 0xFFFF XXXX

Биты 31:9 зарезервированы, их значение должно сохраняться в состоянии сброса.

Биты 8:0 DR[8:0]: Значение данных

Содержит принятый или передаваемый символ данных в зависимости от операции с регистром – чтение или запись.

Регистр данных выполняет двойную функцию (чтение и запись), так как состоит из двух регистров, один для передачи данных (TDR), другой для приема данных (RDR).

Регистр TDR обеспечивает параллельный интерфейс между внутренней шиной и выходным регистром сдвига (см. Рисунок 1). Регистр RDR обеспечивает параллельный интерфейс между входным регистром сдвига и внутренней шиной.

Когда разрешена передача с генерацией бита четности (бит PCE установлен в 1 в регистре USART_CR1), значение, записываемое в СЗБ (бит 7 или бит 8 в зависимости от длины данных) не играет значения, поскольку он заменяется битом четности. Когда на приеме разрешен контроль четности, значение, считываемое из СЗБ, является принятым битом четности.

Регистр скорости передачи (USART_BRR)

Примечание: Счетчики скорости останавливают счет, если биты TE или RE запрещены, соответственно.

Смещение адреса: 0x08

Значение после сброса: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Зарезервировано															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DIV Mantissa[11:0]												DIV Fraction[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Биты 31:16 зарезервированы, их значения должны сохраняться в состоянии сброса.

Биты 15:4 DIV_Mantissa[11:0]: мантисса USARTDIV

Эти 12 бит определяют мантиссу делителя USART (USARTDIV)

Биты 3:0 DIV_Fraction[3:0]: дробная часть USARTDIV

Эти 4 бита определяют дробную часть делителя USART (USARTDIV). Если OVER8=1, бит DIV_Fraction3 не учитывается и должен быть сброшен.

Регистр управления 1 (USART_CR1)

Смещение адреса: 0x0C

Значение после сброса: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Зарезервировано															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OVER8	Зарез- но	UE	M	WAKE	PCE	PS	PEIE	TXEIE	TCIE	RXNEI	IDLEIE	TE	RE	RWU	SBK
rw	Res.	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Биты 31:16 зарезервированы, их значение должно сохраняться в состоянии сброса.

Бит 15 **OVER8**: Режим передискретизации

0: передискретизация на 16

1: передискретизация на 8

Примечание: Передискретизация на 8 недоступна в режимах Smartcard, IrDA и LIN: если SCEN=1, IREN=1 или LINEN=1, тогда OVER8 принудительно сбрасывается аппаратурой в 0.

Бит 14 зарезервирован, значение должно сохраняться в состоянии сброса.

Бит 13 **UE**: Разрешение USART

Сброс данного бита в 0 моментально отключает TX, RX и генератор СК. При этом текущие передачи по TX и RX обрываются (если они идут).

Данный бит устанавливается и сбрасывается программно.

0: передатчик и выходы USART запрещены

1: USART разрешен

Бит 12 **M**: Длина слова

Данный бит определяет длину слова. Данный бит устанавливается и сбрасывается программно.

0: 1 стартовый бит, 8 бит данных, n стоп-бит

1: 1 стартовый бит, 9 бит данных, n стоп-бит

Примечание: Бит M не должен быть модифицирован во время передачи данных (и приема и передачи)

Бит 11 **WAKE**: Метод пробуждения

Данный бит определяет метод пробуждения USART. Данный бит устанавливается и сбрасывается программно.

0: Линия Idle (состояние ожидания на линии)

1: Метка адреса

Бит 10 **PCE**: Разрешение контроля четности

Данный бит выбирает аппаратную генерацию и обнаружение бита четности. Если разрешен контроль бита четности, вычисленное значение четности вставляется на место СЗБ (9 бит, если M=1; 8 бит, если M=0) и на приеме этот бит проверяется на четность.

Данный бит устанавливается и сбрасывается программно.

Как только бит установлен, PCE активируется после текущего байта (при приеме и при передаче).

0: Контроль четности запрещен

1: Контроль четности разрешен

Бит 9 **PS**: Выбор четности/нечетности

Данный бит выбирает, как вычисляется и проверяется бит четности – либо на четность, либо на нечетность (бит PCE установлен). Данный бит устанавливается и сбрасывается программно. Вариант контроля четности/нечетности выбирается после текущего байта.

0: Контроль на четность

1: Контроль на нечетность

Бит 8 **PEIE**: Разрешение прерывание по событию PE

Данный бит устанавливается и сбрасывается программно.

0: прерывание запрещено

1: прерывание генерируется при PE=1 в регистре USART_SR

Бит 7 **TXEIE**: Разрешение прерывание по событию TXE

Данный бит устанавливается и сбрасывается программно.

0: прерывание запрещено

1: прерывание генерируется при TXE=1 в регистре USART_SR

Бит 6 **TCIE**: Разрешение прерывания по завершению передачи

Данный бит устанавливается и сбрасывается программно.

0: прерывание запрещено

1: прерывание генерируется при TC=1 в регистре USART_SR

Бит 5 **RXNEIE**: Разрешение прерывания по событию RXNE

Данный бит устанавливается и сбрасывается программно.

0: прерывание запрещено

1: прерывание генерируется при ORE=1 или RXNE=1 в регистре USART_SR

Бит 4 **IDLEIE**: Разрешение прерывания по событию IDLE

Данный бит устанавливается и сбрасывается программно.

0: прерывание запрещено

1: прерывание генерируется при IDLE=1 в регистре USART_SR

Бит 3 **TE**: Разрешение передатчика

Данный бит разрешает передатчик. Данный бит устанавливается и сбрасывается программно.

0: Передатчик запрещен

1: Передатчик разрешен

Примечание: Во время передачи, нулевой импульс в бите TE ("0" за которым следует "1") передает преамбулу (сигнала ожидания на линии) после текущего слова, кроме режима smartcard.

Когда установлен бит TE, существует 1-битная задержка перед началом передачи.

Бит 2 **RE**: Разрешение приемника

Данный бит разрешает приемник. Данный бит устанавливается и сбрасывается программно.

0: Приемник запрещен

1: Приемник разрешен, и он начинает искать стартовый бит

Бит 1 **RWU**: Пробуждение приемника

Данный бит определяет, находится ли приемопередатчик в режиме молчания или нет.

Данный бит устанавливается и сбрасывается программно, а также может быть сброшен аппаратно при распознавании последовательности пробуждения.

0: приемник в активном режиме

1: приемник в режиме молчания

Примечание: Перед выбором режима молчания (Mute) (установка бита RWU)

приемопередатчик USART должен сначала получить байт данных, иначе он не сможет

работать в режиме молчания с пробуждением по обнаружению сигнала ожидания на линии.

При конфигурации пробуждения по метке адреса (WAKE=1) бит RWU не может быть модифицирован программно, пока установлен бит RXNE.

Бит 0 **SBK**: Отправка сигнала break

Данный бит используется для отправки символов break. Данный бит устанавливается и сбрасывается программно. Он должен быть установлен программой и будет сброшен аппаратно во время стопового бита или символа break.

0: Символ break не передается

1: Символ Break передается

Регистр управления 2 (USART_CR2)

Смещение адреса: 0x10

Значение после сброса: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Зарезервировано															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	STOP[1:0]	CLKEN	CPOL	CPHA	LBCL	Res.	Res.	Res.	Res.	ADD[3:0]				
	-	rw	rw	rw	rw	rw		-	-	rw	rw	rw	rw	rw	rw

Биты 31:15 зарезервированы, их значение должно сохраняться в состоянии сброса.

Бит 14 Зарезервирован

Биты 13:12 **STOP**: стоповые биты

Данные биты используются для программирования стоп бит.

00: 1 стоп-бит

01: 0.5 стоп-бит

10: 2 стоп-бита

11: 1.5 стоп-бита

Бит 11 **CLKEN**: разрешение тактирования

Данный бит позволяет разрешить работу вывода СК.

0: Вывод СК запрещен

1: Вывод СК разрешен

Бит 10 **CPO**L: Полярность тактов

Данный бит позволяет выбрать полярность выхода тактов на выводе СК в синхронном режиме.

Совместно с битом CPHA данный бит определяет взаимосвязь тактов и данных

0: постоянное значение 0 на выводе СК вне окна передачи.

1: постоянное значение 1 на выводе СК вне окна передачи.

Бит 9 **CP**HA: Фаза тактов

Данный бит позволяет выбрать фазу выхода тактов на выводе СК в синхронном режиме.

Совместно с битом CPOL данный бит определяет взаимосвязь тактов и данных (см.

Рисунки 119 и 120)

0: первое изменение сигнала синхронизации является первым перепадом захвата данных

1: второе изменение сигнала синхронизации является первым перепадом захвата данных

Бит 8 **LB**CL: Тактовый импульс последнего бита

Бит позволяет выбрать, будет ли тактовый импульс последнего переданного бита (СЗБ) выдаваться на вывод СК в синхронном режиме.

0: тактовый импульс последнего бита данных не выводится на вывод СК

1: тактовый импульс последнего бита данных выводится на вывод СК

Примечание: 1: Последний бит это 8 или 9 бит данных, в зависимости от выбранного формата, задаваемого битом M в регистре USART_CR1.

Бит 7 зарезервирован, его значение должно сохраняться в состоянии сброса.

Бит 6 Зарезервировано

Бит 5 Зарезервировано

Бит 4 зарезервирован, его значение должно сохраняться в состоянии сброса

Биты 3:0 **ADD[3:0]**: адреса узла USART

Данное битовое поле указывает адрес режима USART.

Данные биты используются в многопроцессорном обмене во время режима молчания, для пробуждения приемопередатчика при обнаружении метки адреса.

Примечание: Эти 3 бита (CPOL, CPHA, LBCL) не должны быть записаны, пока разрешен передатчик.

Регистр управления 3 (USART_CR3)

Смещение адреса: 0x14

Значение после сброса: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Зарезервировано															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Зарезервировано				ONEBIT	CTSIE	CTSE	RTSE	DMAT	DMAR	SCEN	NACK	HDSEL	Res.	Res.	EIE
				rw	rw	rw	rw	rw	rw	rw	rw	rw	-	-	rw

Биты 31:12 зарезервированы, их значение должно сохраняться в состоянии сброса

Бит 11 **ONEBIT**: разрешение режима одной выборки на бит

Бит позволяет выбрать метод оцифровки сигнала. Когда выбран метод одной выборки, флаг обнаружения шума (NF) запрещен.

0: метод трех выборок

1: метод одной выборки

Примечание: функция ONEBIT применима только к битам данных. Это не относится к стартовому биту.

Бит 10 **CTSIE**: Разрешение прерывания CTS

0: прерывание запрещено

1: прерывание генерируется каждый раз при CTS=1 в регистре USART_SR

Бит 9 **CTSE**: Разрешение CTS

0: запрещено аппаратное управление потоком CTS

1: режим CTS разрешен, данные передаются только, когда вход CTS имеет логический 0. Если вход

CTS установлен на лог. 1 во время передачи данных, тогда передача завершается до остановки. Если данные записаны в регистр данных, когда CTS в лог. 1, то передача будет отложена до тех пор, пока на входе CTS не появится логический 0.

Бит 8 **RTSE**: Разрешение RTS

0: запрещено аппаратное управление потоком RTS

1: прерывание RTS разрешено, данные запрашиваются, когда есть место в буфере приемника. Ожидается, что передача данных прекратится после того, как будет передан последний текущий символ данных. Выход RTS имеет логический 0, когда приемопередатчик может принимать данные.

Бит 7 **DMAT**: Разрешение передатчика DMA

Данный бит устанавливается и сбрасывается программно.

1: Режим DMA разрешен для передачи.

0: Режим DMA запрещен для передачи.

Бит 6 **DMAR**: Разрешение приемника DMA

Данный бит устанавливается и сбрасывается программно.

1: Режим DMA разрешен для приема

0: Режим DMA запрещен для приема

Бит 5 **SCEN**: Разрешение режима Smartcard

Данный бит используется для разрешения режима Smartcard.

0: Режим Smartcard запрещен

1: Режим Smartcard разрешен

Бит 4 **NACK**: Разрешение NACK в режиме Smartcard

0: ответ NACK в случае ошибки четности запрещен

1: ответ NACK в случае ошибки четности разрешен

Бит 3 **HDSEL**: Выбор полудуплексного режима

Бит позволяет выбрать однопроводной режим или полудуплексный режим

0: полудуплексный режим не выбран

1: полудуплексный режим выбран

Бит 2 Зарезервировано

Бит 1 Зарезервировано

Бит 0 **EIE**: Разрешение прерывания по ошибке

Бит используется для разрешения генерации прерывания по ошибке фрейма, ошибке переполнения или при флаге шума (FE=1 или ORE=1 или NF=1 в регистре USART_SR) в случае мультибуферного обмена (DMAR=1 в регистре USART_CR3).

0: прерывание запрещено

1: прерывание генерируется каждый раз, когда DMAR=1 в регистре USART_CR3 и FE=1 или

ORE=1 или NF=1 в регистре USART_SR.

Регистр защитного интервала и предделителя (USART_GTPR)

Смещение: 0x18

Значение после сброса: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Зарезервировано															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
GT[7:0]								Зарезервировано			PSC[4:0]				
rw	rw	rw	rw	rw	rw	rw	rw				rw	rw	rw	rw	rw

Биты 31:16 зарезервированы, их значение должно сохраняться в состоянии сброса

Биты 15:8 **GT[7:0]**: значение защитного интервала

Это битовое поле показывает значение защитного интервала времени в единицах тактов скорости.

Биты используются в режиме Smartcard. Флаг завершения передачи устанавливается после этого значения защитного интервала.

Биты 4:0 **PSC[4:0]**: Значение предделителя

– В режиме smartcard:

PSC [4:0]: значение предделителя

Используется для программирования предделителя для деления системной тактовой частоты, чтобы обеспечить тактирование смарт-карты.

Значение, заданное в регистре (5 значащих бит), умножается на 2, чтобы получить коэффициент деления исходной тактовой частоты:

00000: зарезервировано – не программировать это значение

00001: делит источник тактов на 2

00010: делит источник тактов на 4

00011: делит источник тактов на 6

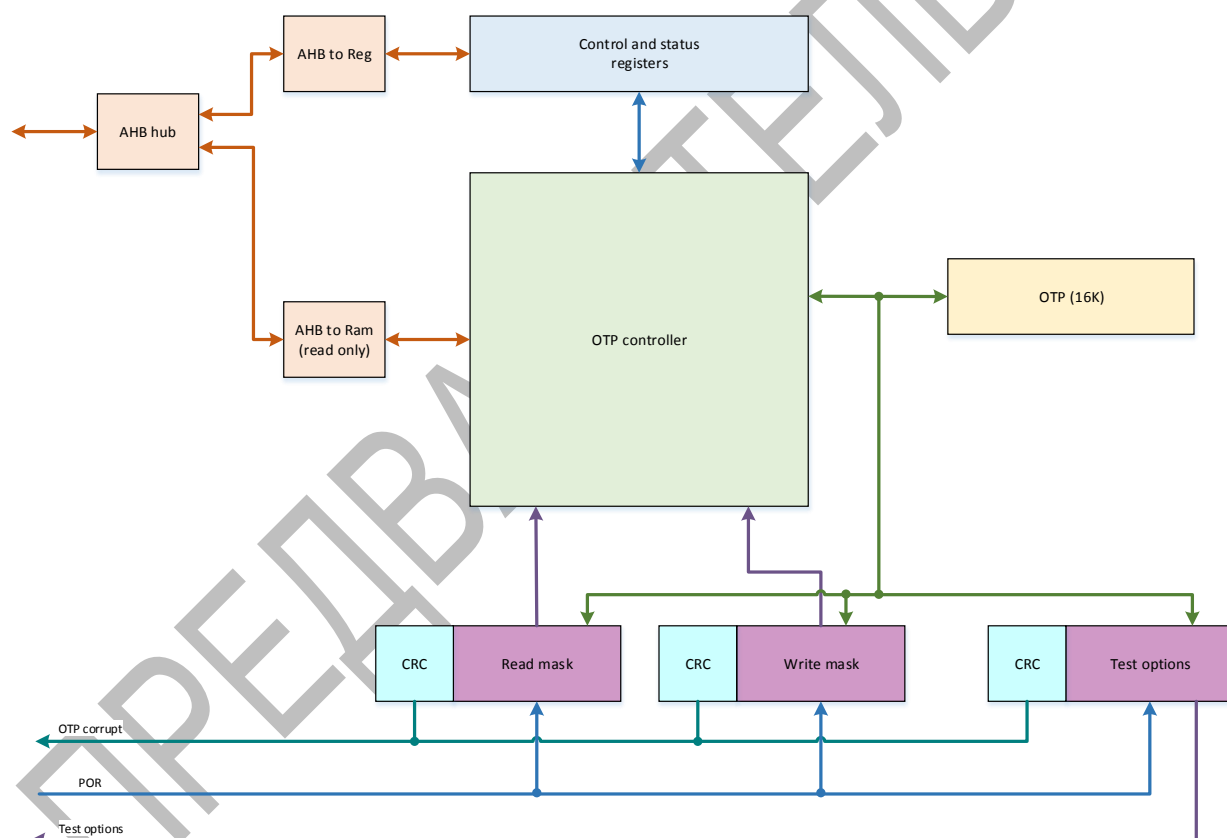
Модуль однократно программируемой памяти (ОТР)

Назначение и свойства

Модуль предназначен для чтения и однократного программирования блока постоянной памяти:

- модуль позволяет читать данные из адресного пространства памяти по АНВ, формируя необходимую задержку сигналом готовности;
- модуль позволяет записывать (программировать) данные через регистровый интерфейс, доступный по АНВ;
- модуль имеет прямое управление блоками памяти (ОТР память) через специальный регистр;
- модуль имеет 1 блок ОТР общим объёмом 128 Кбит (16 Кбайт);
- модуль осуществляет автономную блокировку диагностический функций по данным из специального адреса ОТР;
- модуль имеет вход изоляции блоков ОТР для режима тестирования;
- модуль поддерживает только 32 битную запись в регистры.

Рисунок 128. Структурная схема модуля ОТР.



Карта регистров

Базовый Адрес		Название	Описание
0x7000_0000		OTP_MEM	Адресное пространство однократно программируемой памяти
0x7001_0000		OTP_REG	Блок регистров ОТР
Смещение			

0x0000	0	STAT_CTRL	Регистр статуса и управления
0x0004	1	OTP_STAT	Регистр статуса блоков
0x0008	2	DELAY_0	Регистр задержки 0
0x000c	3	DELAY_1	Регистр задержки 1
0x0010	4	RW_CMD	Регистр команды чтения-записи
0x0014	5	READ_DATA	Регистр считанных данных
0x0018	6	WRITE_PROTECT	Регистр защиты от записи
0x001C	7	READ_PROTECT	Регистр защиты от чтения
0x0020	8	DIR_CTRL	Регистр прямого управления
0x0024	9	DIR_DATA	Регистр данных прямого управления
0x0028	10	TEST_OPT	Регистр включения тестовых опций

Регистр статуса и управления STAT_CTRL (Адрес: 0x00).

бит	сокращенное наименование	функция	доступ
2:0	FSM_STATE	Состояние конечного автомата общего контроллера модуля: 0 - бездействие 1 - инициализация 2 - готовность к работе 3 - ожидание окончания чтения через регистры 4 - ожидание окончания чтения через шину 5 - ожидание окончания записи 6 - режим прямого управления OTP (через специальный регистр)	RO
3	BUSY	OTP выполняет какие-либо действия и не готово принять очередной запрос на чтение или запись	RO
4	OTP_LOCK	Сигнал блокировки OTP от записи, единица в данном сигнале означает что все банки модуля блокированы	RO
5	OTP_PLOCK	Сигнал частичной блокировки OTP от записи, единица в данном сигнале означает что некоторые банки модуля блокированы	RO
6	RW_ERROR	Ошибка чтения или записи Ошибка возникает при попытке чтения или записи во время установленного сигнала BUSY При попытке писать в защищенную от записи область или читать из защищенной от чтения	RO

бит	сокращенное наименование	функция	доступ
		области	
7	TEST_CRC	Состояние сигнала целостности данных специальных тестовых опций	RO
8	WRITE_PCRC	Состояние сигнала целостности данных регистра защиты от записи	RO
9	READ_PCRC	Состояние сигнала целостности данных регистра защиты от чтения	RO
10	DIR_EN	Запрос режима прямого управления	RW
11	DIR_ACK	Подтверждение включения режима прямого управления	RO
12	DIR_ERR	Ошибка включения режима прямого управления Прямое управление нельзя включить: до инициализации OTP блоков (происходит автоматически в начале работы); если есть нарушения целостности какого-либо из специальных регистров; если установлена защита от чтения или записи любого региона	RO
31:13	-	Зарезервировано	RO

Регистр статуса блоков OTP, OTP_STAT_REG (адрес: 0x04).

Бит	Сокращенное наименование	Функция	Доступ
4:0	OTP_0_FSM_STATE	Состояние конечного автомата блока OTP: 0 - бездействие 1 - переход в режим энергосбережения 2 - подготовка сброса 3 - сброс 4 - окончание сброса 5 - активный режим, ожидание чтения-записи 6 - пауза режима чтения после записи 7 - подготовка чтения 8 - чтение 9 - окончание чтения 10 - подготовка записи 11 - сохранение данных для записи 12 - окончание сохранения данных 13 - подготовка программирования 14 - подготовка схемы повышения напряжения 15 - программирование	RO

Бит	Сокращенное наименование	Функция	Доступ
		16 - выключение схемы повышения напряжения 17 - ожидание окончания программирования 18 - ожидание окончания записи 19 - подготовка режима прямого управления 20 - режим прямого управления	
5	-	Зарезервировано	RO
6	OTP_0_LOCK	Состояние блокировки записи модуля OTP 0	RO
7	OTP_0_BUSY	Модуль OTP 0 выполняет какие-либо действия и не готово принять очередной запрос на чтение или запись	RO
31:8	-	Зарезервировано	RO

Регистр задержки 0, DELAY_0_REG (адрес: 0x08).

Бит	Сокращенное наименование	Функция	Доступ
3:0	DELAY_20NS	Параметр задержки для формирования паузы 20 нс. Величина задержки задается в тактах основной частоты работы модуля, пауза равна N+1 такту, где N заданное в поле число. И должна быть не меньше 20 нс.	RW
7:4	DELAY_70NS	Параметр задержки для формирования паузы 70 нс при чтении данных из OTP по шине АНВ. Величина задержки задается в тактах основной частоты работы модуля, пауза равна N+1 такту, где N заданное в поле число. И должна быть не меньше 70 нс.	RW
12:8	DELAY_50NS	Параметр задержки для формирования паузы 50 нс. Величина задержки задается в тактах основной частоты работы модуля, пауза равна N+1 такту, где N заданное в поле число. И должна быть не меньше 50 нс.	RW
15:13	-	Зарезервировано	RO
24:16	DELAY_01US	Параметр задержки для формирования паузы 1 мкс. Величина задержки задается в тактах основной частоты работы модуля, пауза равна N+1 такту, где N заданное в поле число. И должна быть не меньше 1 мкс и не более 2 мкс.	RW
31:25	-	Зарезервировано	RO

Регистр задержки 1, DELAY 1 REG (адрес: 0x0C).

Бит	Сокращенное наименование	Функция	доступ
16:0	DELAY_16US	Параметр задержки для формирования паузы 16 мкс. Величина задержки задается в тактах основной частоты работы модуля, пауза равна N+1 такту, где N заданное в поле число. И должна быть не меньше 16 мкс и не больше 17 мкс.	RW
31:17	-	Зарезервировано	RO

Регистр команды чтения-записи RW CMD REG (адрес: 0x10).

Бит	Сокращенное наименование	Функция	Доступ
16:0	ADDR	Битовый адрес записи и чтения. Запись производится битами, чтение производится байтами, при чтении младшие 3 бита игнорируются и считаются равными 0.	RW
23:17	-	Зарезервировано	RO
24	DATA_0	Данные для записи в ОТР, 0 - не производить запись. Начальное состояние ОТР - 0, записать можно только 1, теперь 1 уже нельзя	RW
25	-	Зарезервировано	RO
26	-	Зарезервировано	RO
27	-	Зарезервировано	RO
28	READ	Команда на чтение из указанного адреса, при наличии 1 в данном поле и 0 в поле команды записи производится процедура чтения. Если выбранная область не защищена от чтения и ОТР не занято, произойдет чтение данных, считанные данные будут возвращены в регистр считанных данных. Окончание контролируется по флагу BUSY в регистре статуса.	RW
29	WRITE	Команда на запись в указанный адрес, при наличии 1 в данном поле и 0 в поле команды чтения производится процедура записи. Если выбранная область не защищена от записи ОТР не занято, произойдет запись данных. Окончание контролируется по флагу BUSY в регистре статуса.	RW

Бит	Сокращенное наименование	Функция	Доступ
30	-	Зарезервировано	RO
31	RW_ERROR	Ошибка выполнения последней операции чтения или записи Ошибка возникает при попытке чтения или записи во время установленного сигнала BUSY При попытке писать в защищенную от записи область или читать из защищенной от чтения области	RO

Регистр считанных данных READ_DATA_REG (адрес: 0x14).

Бит	Сокращенное наименование	Функция	Доступ
7:0	DATA_0	Данные считанные из 0 банка OTP в случае успешного завершения процедуры чтения, при ошибках в поле заноситься константа 0xEF	RO
15:8	-	Зарезервировано	RO
23:16	-	Зарезервировано	RO
31:24	-	Зарезервировано	RO

Регистр защиты от записи WRITE_PROTECT_REG (адрес: 0x18).

Бит	Сокращенное наименование	Функция	Доступ
7:0	-	Каждый бит регистра отвечает за область памяти в 2048 байт. 0 бит за адреса 0 – 2047, 1 бит за адреса 2048 – 4095 и так далее. 1 в бите запрещает запись в данный регион памяти. Биты защиты можно только установить, сбросить биты защиты можно только общим сбросом по питанию. Сброс процессора через регистры управления не снимает установленные биты. При штатной записи данных в регистр, автоматически рассчитывается контрольная сумма для нового значения. По этой сумме контролируется целостность данных, на случай модификаций бит защиты сторонними методами.	RW
31:8	-	Зарезервировано	RO

Регистр защиты от чтения READ_PROTECT_REG (адрес: 0x1C).

Бит	Сокращенное наименование	Функция	Доступ
7:0	-	Каждый бит регистра отвечает за область памяти в 2048 байт. 0 бит за адреса 0 - 2047, 1 бит за адреса 2048 - 4095 и так далее. 1 в бите запрещает чтение из данного региона памяти. При попытке чтения из защищенного региона возвращается значение 0xEF. Биты защиты можно только установить, сбросить биты защиты можно только сбросом процессора. При штатной записи данных в регистр, автоматически рассчитывается контрольная сумма для нового значения. По этой сумме контролируется целостность данных, на случай модификаций бит защиты сторонними методами.	RW
31:8	-	Зарезервировано	RO

Регистр включения тестовых опций TEST_OPT_REG (адрес: 0x28).

Бит	Сокращенное наименование	Функция	Доступ
7:0	DEBUG	Регистр содержит биты разрешения включения отладчика. Если не все биты данного поля будут единицами или будет нарушена целостность данных регистра, отладчик будет заблокирован. Регистр обновляется при любом чтении из последнего адреса ОТР, в регистр заноситься инверсия считанного значения из ОТР значения. При штатном обновлении данных в регистр, автоматически рассчитывается контрольная сумма для нового значения. По этой сумме контролируется целостность данных, на случай модификаций сторонними методами. Сброс регистра происходит общим сбросом питания, сброс процессора через регистры управления не изменяет регистр. В сброшенном состоянии все опции отключены.	RO
31:8	ATPG	Регистр содержит биты разрешения включения скановых цепочек криптоблока. Если не все биты данного поля будут единицами или будет нарушена целостность данных регистра, скановые цепочки будут заблокированы, тестирование через ATPG будет невозможно. Регистр обновляется при любом чтении из последнего адреса ОТР, в регистр заноситься инверсия считанного значения из ОТР значения. При штатном обновлении данных в регистр, автоматически рассчитывается контрольная сумма	

Бит	Сокращенное наименование	Функция	Доступ
		для нового значения. По этой сумме контролируется целостность данных, на случай модификаций сторонними методами. Сброс регистра происходит общим сбросом питания, сброс процессора через регистры управления не изменяет регистр. В сброшенном состоянии все опции отключены.	

ПРЕДВАРИТЕЛЬНО

Предельно-допустимые характеристики микросхемы

Таблица 465 Предельно-допустимые и предельные режимы эксплуатации микросхем

Наименование параметра, единица измерения	Буквенное обозначени е параметра	Норма параметра			
		Предельно- допустимый режим		Предельный режим	
		не менее	не более	не менее	не более
Напряжение источника питания, В	U _{CC}	2,2	3,6	–	3,9
Напряжение источника питания при использовании АЦП, В	U _{CCA}	3,0	3,6		3,9
Напряжение источника питания батарейного домена, В	U _{CCB}	1,8	3,6	–	3,9
Входное напряжение высокого уровня, В,	U _{IH}	2,0	U _{CC}	–	U _{CC} +0,3
Входное напряжение низкого уровня, В,	U _{IL}	0	0,8	–0,3	–
Выходной ток высокого уровня, мА, при работе в цифровом режиме на выводах: PA, PB, PC	I _{OH}	–4	–	–6	–
Выходной ток низкого уровня, мА на выводах: PA, PB, PC	I _{OL}	–	4	–	6
Частота следования импульсов тактовых сигналов, МГц	f _C	–	60	–	–
Частота следования импульсов тактовых сигналов HSE, МГц при BYPASS=0 при BYPASS=1	f _{C_HSE}	8	16	–	–
		–	60		
Частота следования импульсов тактовых сигналов LSE, кГц при BYPASS=0 при BYPASS=1	f _{C_LSE}	32	33	–	–
		–	1 000		
Частота следования импульсов тактовых сигналов PLL, МГц	f _{C_PLL}	8	16	–	–
Емкость нагрузки, пФ, на выводах: PA, PB, PC	C _L	–	30	–	–
Время хранения информации, лет, при T=25 °C при T=85 °C	t _{GS}	25	–	–	–
		10	–	–	–
Параметры АЦП					
Напряжение нижней границы опоры АЦП, В	U _{ADC1_REF-}	0	–	–	–
Напряжение верхней границы опоры АЦП, В	U _{ADC0_REF+}	U _{REF(ADC)}	U _{CCA}	–	–

Наименование параметра, единица измерения	Буквенное обозначени е параметра	Норма параметра			
		Предельно- допустимый режим		Предельный режим	
		не менее	не более	не менее	не более
Диапазон напряжения внешнего опорного источника АЦП, В $U_{REF(ADC)}$	$U_{REF(ADC)}$	2,3	2,5	–	–
Диапазон напряжения на входе АЦП, В	U_{AIN}	U_{ADC1_REF-}	U_{ADC0_REF+}	–0,3	3,9
Частота следования импульсов тактовых сигналов АЦП, МГц	$f_{C_ADC_S}$	–	28	–	–
Параметры $\Sigma\Delta$АЦП					
Частота следования импульсов тактовых сигналов $\Sigma\Delta$ АЦП, МГц	$f_{C_ADC_D}$	–	8,192	–	–
Амплитуда входного дифференциального сигнала $\Delta\Sigma$ АЦП, В	A_{NADC_D}	–	1	–	–
Напряжение на входе $\Sigma\Delta$ АЦП, В	U_{IADC_D}	–0,5	0,5	–0,8	$U_{CC}+0,3$
Пр и м е ч а н и е – Не допускается одновременное задание двух предельных режимов					

Электрические параметры микросхемы

Таблица 466 Электрические параметры микросхем при приёмке и поставке

Наименование параметра, единица измерения, режим измерения	Буквенное обозна- чение пара- метра	Норма параметра		Температура среды, °C
		не менее	не более	
Выходное напряжение высокого уровня, В, на выводах PA, PB, PC, PD	U _{OH}	2,4	–	25, 85, – 50
Выходное напряжение низкого уровня, В, на выводах PA, PB, PC, PD	U _{OL}	–	0,4	
Входной ток утечки высокого уровня, мкА, (при работе в цифровом режиме) на выводах: PA, PB, PC, PD, nRESET, WAKEUP	I _{ILH}	– 1,0	1,0	
на выводе: JTAG_EN		– 180	180	
на выводе OSC_IN		– 40	40	
Ток утечки низкого уровня цифровых входов, мкА, (при работе в цифровом режиме) на выводах: PA, PB, PC, PD, RESET, WAKEUP, JTAG_EN	I _{ILL}	– 1	1	25, 85, – 50
на выводе OSC_IN		– 40	40	
Статический ток потребления, мкА при выключенном стабилизаторе напряжения	I _{CCS}	–	10	25, 85, – 50
при включенном стабилизаторе напряжения		–	100	
Динамический ток потребления, мА, при f _c ≤60 МГц	I _{OCC}	–	70	25, 85, – 50
Выходная частота HSI RC-генератора, МГц (после подстройки)	f _{O_HSI}	7,9	8,1	25, 85, – 50
Выходная частота LSI RC-генератора, кГц (после подстройки)	f _{O_LSI}	30	32	
Выходная частота PLL, МГц	f _{O_PLL}	60	–	25, 85, – 50
максимальная		–	8	
минимальная				
Параметры АЦП последовательного приближения				
Разрядность АЦП	E _{NADC}	10	–	25, 85, – 50
Дифференциальная нелинейность, единица младшего разряда	E _{DLADC}	–1	2	
Интегральная нелинейность, единица младшего разряда	E _{ILADC}	–2	2	
Ошибка смещения, единица младшего разряда	E _{OFFADC}	–3	3	
Ошибка усиления, %	E _{GAINADC}	–0,5	0,5	
Параметры ΣΔ АЦП				
Выходное напряжение VR_2p4V, В	U _{OBIAS}	2,4	2,5	25,

Наименование параметра, единица измерения, режим измерения	Буквенное обозна- чение пара- метра	Норма параметра		Температура среды, °C
		не менее	не более	
Соотношение сигнал / шум, дБ усиление 0дБ, $A_{NADC_D} = 1B$, $f_{NADCO} = 4КГц$	SNR_D_0	77	—	85, — 50
Соотношение сигнал / шум, дБ усиление +6дБ, $A_{NADC_D} = 0,5B$, $f_{NADCO} = 4КГц$	SNR_D_6	74	—	
Соотношение сигнал / шум, дБ усиление +12дБ, $A_{NADC_D} = 0,25B$, $f_{NADCO} = 4КГц$	SNR_D_12	71	—	
Соотношение сигнал / шум, дБ усиление +18дБ, $A_{NADC_D} = 0,125B$, $f_{NADCO} = 4КГц$	SNR_D_18	70	—	
Ошибка усиления предусилителя, дБ	GAIN_ERR	—	0,25	

Габаритный чертеж микросхемы

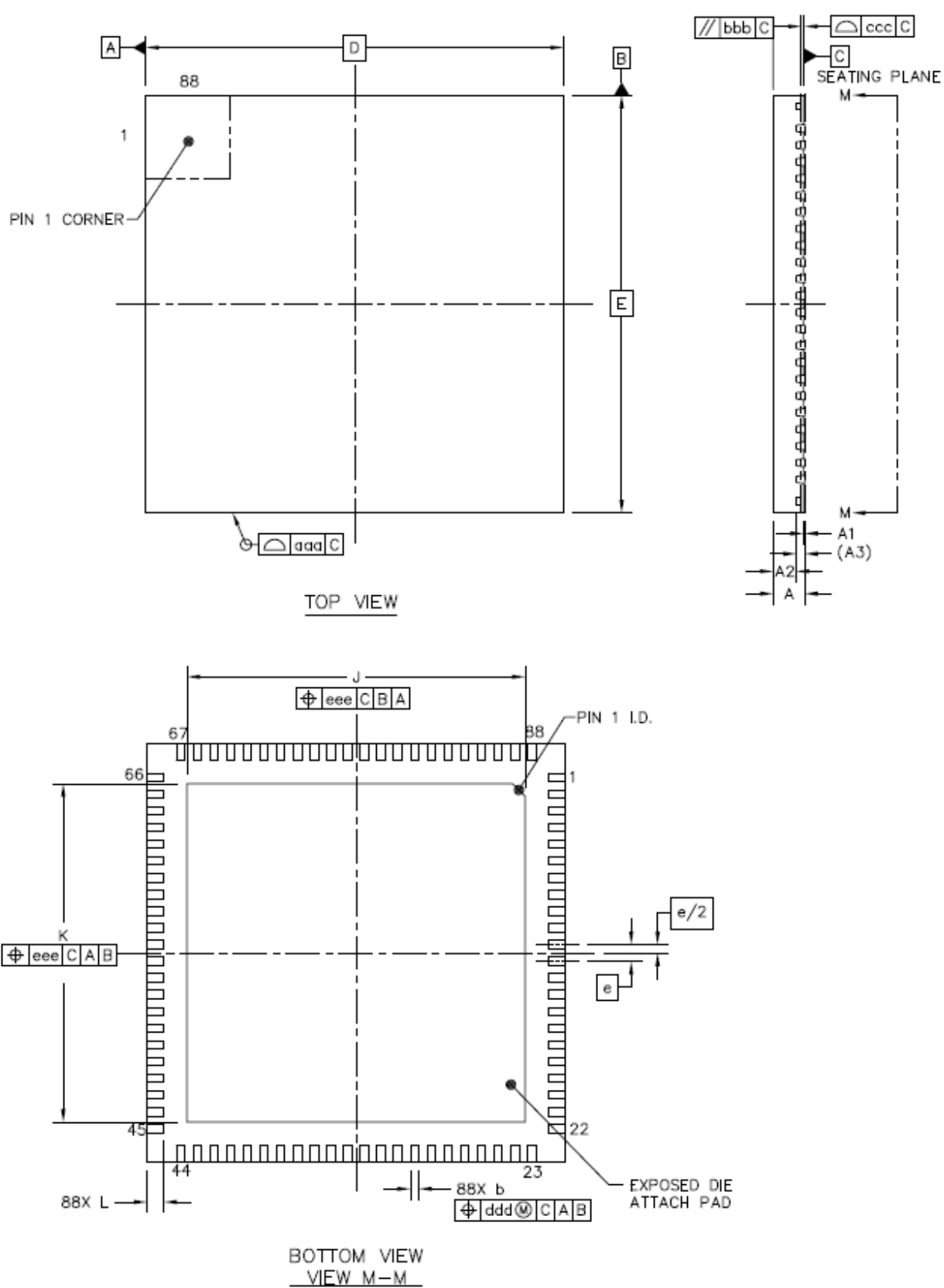


Рисунок 129. Корпус 88 LD QFN-EP

		Обозначение	Минимум	Типичное	Максимум
Общая толщина		A	0.7	0.75	0.8
Расстояние от края		A1	0	0.035	0.05
Толщина формы		A2	-	0.55	0.57
Толщина крышки		A3	0.203 REF		
Ширина вывода		b	0.15	0.2	0.25
Размер корпуса	X	D	10 BSC		
	Y	E	10 BSC		
Шаг выводов		e	0.4 BSC		
Размер колодца	X	J	8	8.1	8.2
	Y	K	8	8.1	8.2
Длина выводов		L	0.35	0.4	0.45
Отклонение края корпуса		aaa	0.1		
Неравномерность формы		bbb	0.1		
Компланарность		ccc	0.08		
Смещение вывода		ddd	0.1		
Смещение колодца		eee	0.1		

Информация для заказа

Обозначение	Маркировка	Тип корпуса	Температурный диапазон
K1986BK025	MDR32F02FI	88 LD QFN-EP	минус 50 – 85 °C

ПРЕДВАРИТЕЛЬНО

Лист регистрации изменений

№ п/п	Дата	Версия	Краткое содержание изменения	№№ изменяемых листов
49	13.10.2020	1.49	Скорректировано описание регистра mtvec	
48	07.10.2020	1.48	Добавлено описание раздела UART (ISO7816)	
47	23.09.2020	1.47	Скорректировано согласно замечаниям Козлова А.	
46	10.09.2020	1.46	Скорректировано по результатам совещания.	
45	08.09.2020	1.45	Добавлено описание работы сторожевых таймеров.	
44	04.09.2020	1.44	Скорректировано согласно замечаниям Костромина И.С.	
43	17.08.2020	1.43	Скорректирована структурная схема подачи питания	
42	07.08.2020	1.42	Дополнен раздел “Прерывания” описанием регистров для работы с прерываниями	
41	20.07.2020	1.41	Корректировка по замечаниям разработчиков аналоговых блоков	
40	23.06.2020	1.40	Корректировка по замечаниям пользователей	
39	06.05.2020	1.39	Изменена маркировка корпуса	
38	02.04.2020	1.38	Поправлен раздел SENSORS	
37	27.03.2020	1.37	Поправлена опечатка для бит PLSB	
36	19.03.2020	1.36	Добавлено прерывание от блока OTP	
35	18.03.2020	1.35	Изменена распиновка выводов	
34	12.03.2020	1.34	Изменено описание блока детектора напряжения питания	
33	10.03.2020	1.33	Изменена логика работы сигнала управления внешним ключом аварийного питания. Добавлен бит программного выключения ключа. Добавлены биты состояния сигналов WAKEUP.	
32	10.03.2020	1.32	Изменено описание UART интерфейса	
31	10.03.2020	1.31	Изменена таблица электрических параметров микросхемы	
30	05.03.2020	1.30	Изменено описание SAR ADC	
29	27.02.2020	1.29	Добавлен дополнительный вывод питания в корпусе	
28	17.02.2020	1.28	В разделе описания батарейного домена добавлена логика работы вывода SW	
27	14.02.2020	1.27	Удалено описание режима standby	
26	13.02.2020	1.26	Добавлен новый вывод SW	
25	31.01.2020	1.25	Удалены разделы Сопроцессор длинночисленной арифметики и Генератор шума	
24	30.01.2020	1.24	Исправлен раздел Контроллер АЦП	
23	22.01.2020	1.23	Исправлен раздел Блок АЦП для	

			измерения напряжений и токов в электрической сети	
22	21.01.2020	1.22	Исправлен раздел Встроенный RC генератор LSI.	
21	20.01.2020	1.21	Добавлен новый биты включения аварийного питания и бит включения генератора шума по питанию	
20	17.01.2020	1.20	Добавлены новые биты управления в батарейном домене	
19	06.12.2019	1.19	Скорректировано описание контроллера OTP	
18	02.12.2019	1.18	Изменен генератор случайных чисел	
17	19.11.2019	1.17	Изменена цикловка в обоих корпусах	
16	12.11.2019	1.16	Добавлен новый корпус	
15	07.11.2019	1.15	Изменено описание раздела “Порты ввода-вывода”	
14	06.11.2019	1.14	Изменено описание раздела “Детектор напряжения питания”	
13	05.11.2019	1.13	Добавлен UART4. Изменено назначение выводов.	
12	29.10.2019	1.12	Добавлены тестовые выходы управления в регистры АЦП.	
11	28.10.2019	1.11	Изменен батарейный домен и его входные/выходные сигналы.	
10	10.10.2019	1.10	Изменено адресное пространство шины TCM.	
9	31.07.2019	1.9	Изменена нумерация выводов в корпусе.	
8	10.07.2019	1.8	Изменён тип корпуса.	
7	25.06.2019	1.7	Увеличена разрядность регистров FxPER_FREQ. Добавлены младшие 25 разрядов для мощностей сигма-дельта АЦП в отдельные регистры. Введён регистр IRMS неактивного канала.	
6	06.06.2019	1.6	Добавлен раздел с описанием OTP контроллера	
5	04.06.2019	1.5	Скорректировано описание криптографических блоков после тестирования	
4	23.04.2109	1.4	Добавлена дополнительная частота для сторожевого таймера WWDG	
3	18.04.2019	1.3	Добавлен бит FMT в регистре BKP_RTC	
2	16.04.2019	1.2	Изменены разделы “Батарейный домен и часы реального времени” и “Детектор напряжения питания”	
1	08.04.2019	1.1	Введена впервые	