

17 ИНТЕРФЕЙС ВНЕШНЕЙ ШИНЫ

Интерфейс внешней шины (EBIU, External Bus Interface Unit) обеспечивает интерфейс с устройствами внешней памяти, не требующий дополнительных компонентов. Процессор поддерживает синхронные DRAM (SDRAM) и совместим со стандартами SDRAM PC100 и PC133. EBIU также поддерживает асинхронные устройства памяти, такие как SRAM, ROM, FIFO, флэш-память и схемы на базе ASIC/FPGA.

Обзор

EBIU обслуживает запросы на обращение к внешней памяти, поступающие от ядра или канала DMA. Приоритет запросов определяется контроллером внешней шины. В зависимости от адреса запрос обслуживается контроллером SDRAM EBIU или контроллером асинхронной памяти EBIU.

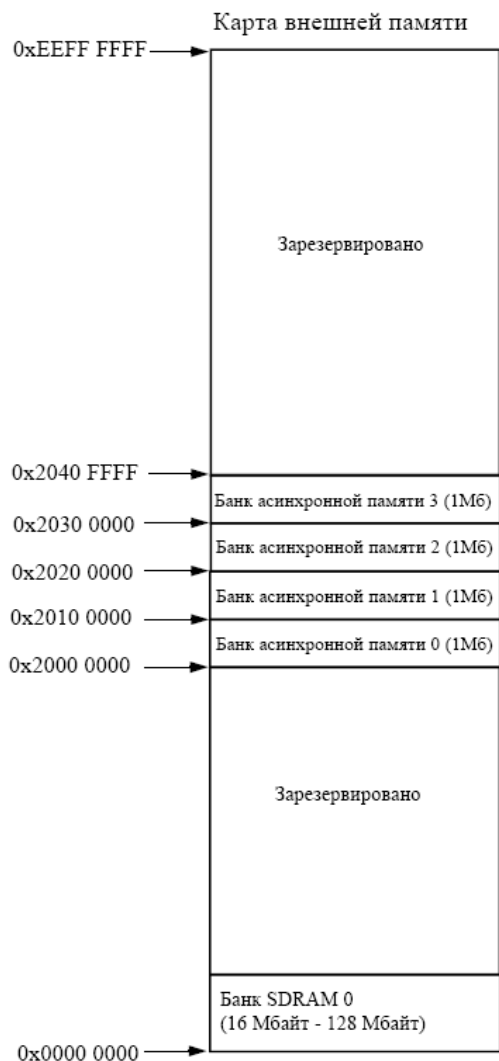
EBIU тактируется тактовым сигналом системы (SCLK). Все устройства синхронной памяти, подключаемые к процессору, работают с частотой SCLK. Отношение частоты ядра к частоте SCLK задаётся программно при помощи регистров схемы фазовой автоподстройки частоты (PLL), отображённых в карте памяти. Дополнительную информацию см. в разделе “Управление отношением тактовой частоты ядра к тактовой частоте системы” главы 8.

Пространство внешней памяти показано на рис. 17-1. Одна область выделена для поддержки SDRAM. Временные параметры интерфейса SDRAM и размер области SDRAM изменяются программно. Размер пространства SDRAM может изменяться в диапазоне от 16 до 128 Мбайт. Начальный адрес пространства памяти SDRAM – 0x0000 0000. Область от конца пространства памяти SDRAM до адреса 0x2000 0000 зарезервирована.

Следующие четыре области выделены для поддержки устройств асинхронной памяти. Параметры каждой области асинхронной памяти настраиваются независимо. Это позволяет процессору поддерживать устройства памяти с различными характеристиками. EBIU имеет отдельные выходные выводы выбора микросхем памяти для каждой области асинхронной памяти.

Следующая область является зарезервированной. Обращения к этой области не вызывают выполнения транзакций внешней шины: операции записи не влияют на значения внешней памяти, при чтении возвращаются неопределённые значения. При этом EBIU генерирует отклик ошибки на внутренней шине, который, в свою очередь, вызывает аппаратное исключение доступа ядра или (необязательно) прерывание канала DMA.

17 Интерфейс внешней шины



Примечание: Все области памяти, не отмеченные как зарезервированные, могут адресоваться и ядром и системой

Рис. 17-1. Карта внешней памяти

Блок-схема

На рис. 17-2 показана концептуальная блок-схема EBIU и его интерфейсов. Названия сигналов, имеющих активный низкий уровень, сопровождаются чертой сверху.

Так как в один момент времени может выполняться обращение только к одному внешнему устройству памяти, управляющие, адресные выходы и выходы данных всех устройств мультиплексируются на выводах процессора. Арбитраж совместно используемых выводов осуществляется контроллером асинхронной памяти (АМС,

17 Интерфейс внешней шины

Asynchronous Memory Controller) и контроллером SDRAM (SDC, SDRAM Controller)

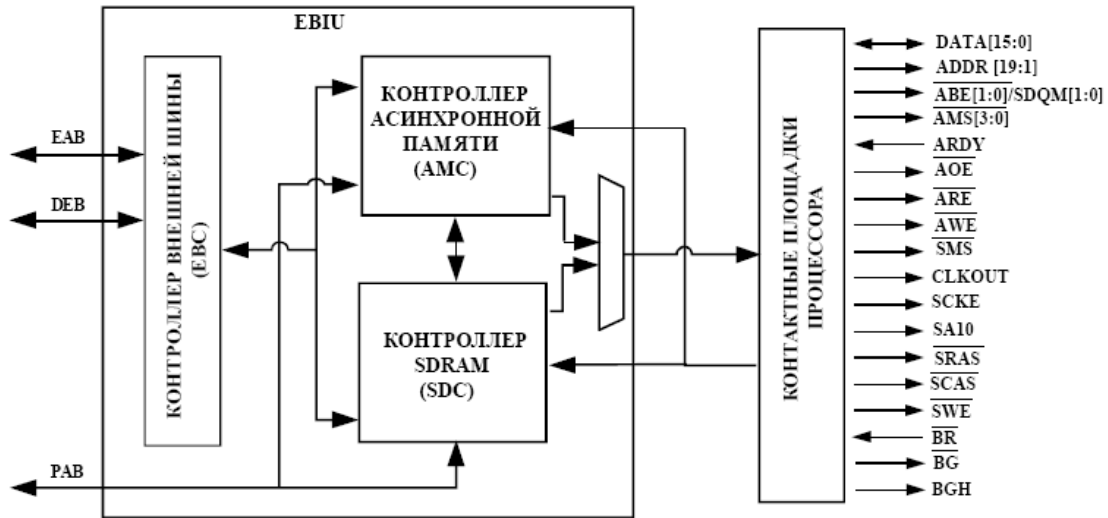


Рис. 17-2. Устройство интерфейса внешней шины (EBIU)

Интерфейсы внутренней памяти

EBIU функционирует как ведомое устройство на трёх внутренних шинах процессора:

- шине внешнего доступа (EAB), управляемой устройством управления памятью ядра посредством запросов внешней шины ядром;
- внешней шине DMA (DEB), управляемой контроллером DMA посредством запросов внешней шины, поступающих от любого канала DMA;
- шине периферийного доступа (PAB), управляемой ядром посредством запросов доступа к регистрам системы, отображённым в карте памяти.

Интерфейсы с этими шинами, а также регистры EBIU и регистры ввода/вывода, соединённые с контактными площадками, являются синхронными и работают с частотой SCLK. Доступ к асинхронной внешней памяти и к внешней синхронной DRAM обеспечивается шиной EAB. В зависимости от внутреннего адреса, используемого при доступе к EBIU, доступом к внешней памяти управляет либо AMC, либо SDC. Так как AMC и SDC имеют общий интерфейс с внешними выводами, доступ выполняется последовательно. Арбитраж доступов выполняется на основании запросов EAB.

Третья шина (PAB) используется только для доступа к регистрам управления и состояния EBIU, отображённым в карте памяти. PAB имеет отдельные соединения с AMC и SDC; арбитраж или добавление циклов ожидания при одновременных доступах к EBIU по шинам PAB и EAB не требуется.

Арбитраж запросов доступа к внешней памяти, поступающих по шинам EAB и DEB, должен осуществлять контроллер внешней шины (EBC, External Bus Controller). EBC направляет запросы чтения и записи в соответствующий

17 Интерфейс внешней шины

контроллер памяти, в зависимости от адреса. AMC и SDC конкурируют за доступ к совместно используемым ресурсам логики управления контактными площадками. Разрешение конфликтных ситуаций выполняется по конвейерному принципу арбитром EBC. В большинстве случаев транзакции, инициируемые ядром, имеют приоритет перед доступами в режиме DMA. Однако, если контроллер DMA обнаруживает избыточность транзакций ядра, он может запросить временное увеличение приоритета по сравнению с ядром.

Интерфейсы внешней памяти

AMD и SDC совместно используют выводы адреса и данных внешнего интерфейса, а также некоторые управляющие сигналы. Общие выводы перечислены ниже:

- ADDR [19:1], шина адреса;
- DATA [15:0], шина данных;
- $\overline{\text{ABE}}[1:0]/\text{SDQM}[1:0]$, сигналы разрешения передачи байтов AMC/маскирования данных SDC;
- $\overline{\text{BR}}$, $\overline{\text{BG}}$, $\overline{\text{BGH}}$, сигналы управления доступом к внешней шине.

Мультиплексирование остальных сигналов не выполняется.

В таблицах 17-1 и 17-2 описываются сигналы, связанные с каждым из интерфейсов.

Таблица 17-1. Сигналы интерфейса асинхронной памяти

Контакт	Тип вывода ¹	Описание
DATA [15:0]	I/O	Внешняя шина данных
ADDR [19:1]	O	Внешняя шина адреса
$\overline{\text{AMS}}[3:0]$	O	Выбор микросхемы асинхронной памяти
$\overline{\text{AWE}}$	O	Разрешение записи в асинхронную память
$\overline{\text{ARE}}$	O	Разрешение чтения из асинхронной памяти
$\overline{\text{AOE}}$	O	Разрешение выдачи выходных сигналов асинхронной памяти. В большинстве случаев вывод $\overline{\text{AOE}}$ должен подключаться к выводу $\overline{\text{OE}}$ внешнего асинхронного устройства, отображённого в карте памяти. Для определения сигнала интерфейса, который следует использовать в проекте, см. информацию о временных параметрах сигналов $\overline{\text{ARE}}$ и $\overline{\text{AOE}}$ в <i>ADSP-BF531/ADSP-BF532/ADSP-BF533 Embedded Processor Data Sheet</i> .
ARDY	I	Отклик готовности асинхронной памяти. Следует отметить, что этот вход является синхронным.
$\overline{\text{ABE}}[1:0]/\text{SDQM}[1:0]$	O	Разрешение передачи байтов.

¹ Типы выводов: I = Вход, O = Выход

Таблица 17-2. Сигналы интерфейса SDRAM

17 Интерфейс внешней шины

Контакт	Тип вывода ¹	Описание
DATA[15:0]	I/O	Внешняя шина данных
ADDR[19:18], ADDR[16:1]	O	Внешняя шина адреса Соединяется с адресными выводами SDRAM. Адрес банка выдаётся на выводах ADDR[19:18]; эти выводы должны подключаться к выводам BA[1:0] SDRAM.
$\overline{\text{SRAS}}$	O	Вывод stroba адреса строки Соединяется с выводом $\overline{\text{RAS}}$ SDRAM
$\overline{\text{SCAS}}$	O	Вывод stroba адреса столбца Соединяется с выводом $\overline{\text{CAS}}$ SDRAM
$\overline{\text{SWE}}$	O	Вывод разрешения записи в SDRAM Соединяется с выводом $\overline{\text{WE}}$ SDRAM
$\overline{\text{ABE}}[1:0]/\overline{\text{SDQM}}[1:0]$	O	Выводы маскирования данных SDRAM Соединяются с выводами $\overline{\text{DQM}}$ SDRAM
$\overline{\text{SMS}}$	O	Вывод выбора внешнего банка памяти SDRAM Соединяется с выводом $\overline{\text{CS}}$ (выбор микросхемы) SDRAM. Активный уровень – низкий.
SA10	O	Вывод A10 SDRAM Этот вывод используется интерфейсом SDRAM для выполнения регенерации в моменты времени, когда шина используется контроллером асинхронной памяти. Соединяется с выводом A[10] SDRAM.
SCKE	O	Вывод разрешения тактового сигнала SDRAM Соединяется с выводом $\overline{\text{CKE}}$ SDRAM.
CLKOUT	O	Вывод выхода тактового сигнала SDRAM Сигнал на этом выводе изменяет состояние с частотой тактового сигнала системы. Соединяется с выводом $\overline{\text{CLK}}$ SDRAM.

¹ Типы выводов: I = Вход, O = Выход

Модель программирования EBIU

В этом разделе описывается модель программирования EBIU. Она основана на использовании регистров системы, отображённых в карте памяти.

EBIU содержит шесть регистров управления и один регистр состояния:

- регистр глобального управления асинхронной памятью (EBIU_AMGCTL),
- регистр управления банками асинхронной памяти 0 (EBIU_AMBCTL0),
- регистр управления банками асинхронной памяти 1 (EBIU_AMBCTL1),
- регистр глобального управления SDRAM (EBIU_SDGCTL),
- регистр управления банком памяти SDRAM (EBIU_SDBCTL),
- регистр управления частотой регенерации SDRAM (EBIU_SDRRC),
- регистр состояния управления SDRAM (EBIU_SDSTAT).

Каждый из перечисленных регистров подробно описывается в разделах этой главы, посвящённых AMC и SDC.

17 Интерфейс внешней шины

Обнаружение ошибок

EBIU отвечает на любые операции шины, адресующие диапазон памяти 0x0000 0000 – 0xEEFF FFFF, даже если они адресуют зарезервированную или неактивную область памяти. Отклик EBIU заключается в завершении операции шины (выставление необходимого количества подтверждений в соответствии с требованиями ведущего устройства шины) и установлении активного уровня сигнала ошибки шины при возникновении следующих ошибок:

- любое обращение к зарезервированному пространству внешней памяти;
- любое обращение к неактивному банку внешней памяти;
- любое обращение к незанятой области банка SDRAM.

Если ядро запрашивает выполнение операции, вызывающей ошибку, отклик ошибки шины, формируемый EBIU, преобразовывается во внутреннее прерывание ядра HWE (это прерывание может маскироваться ядром). Если операция, вызывающая ошибку, запрашивается ведущим устройством DMA, отклик ошибки шины захватывается соответствующим контроллером DMA, который может (необязательно) генерировать прерывание ядра.

Интерфейс асинхронной памяти

Интерфейс асинхронной памяти позволяет реализовать подключение различных типов устройств памяти и периферийных устройств без использования дополнительных компонентов. К ним относятся SRAM, ROM, FIFO, флэш-память и схемы на базе ASIC/FPGA. Поддерживаются четыре области асинхронной памяти, каждой из которых соответствует уникальный сигнал выбора, показанный в таблице 17-3.

Таблица 17-3. Диапазоны адресов банков асинхронной памяти

Сигнал выбора банка памяти	Начальный адрес	Конечный адрес
$\overline{\text{AMS}}[3]$	2030 0000	203F FFFF
$\overline{\text{AMS}}[2]$	2020 0000	202F FFFF
$\overline{\text{AMS}}[1]$	2010 0000	201F FFFF
$\overline{\text{AMS}}[0]$	2000 0000	200F FFFF

Дешифрация адресов асинхронной памяти

Каждый банк асинхронной памяти имеет фиксированный размер – 1 Мбайт; однако, активные банки не обязательно заполнять полностью. В отличие от SDRAM, где может потребоваться поддержка больших структур памяти, покрывающих несколько банков, тип кода и данных, хранимых в асинхронной памяти, позволяет структурировать их так, чтобы они занимали один из поддерживаемых банков асинхронной памяти.

17 Интерфейс внешней шины



Необходимо отметить, что обращения к незаполненным или частично заполненным банкам АМС не приводят ошибкам шины; при этом выполняется чтение или запись по альтернативным достоверным адресам АМС.

Сигналы асинхронной памяти перечислены в таблице 17-1. Их параметры настраиваются программно, за счёт чего достигается гибкость интерфейса с устройствами, работающими с различными скоростями. Примеры интерфейсов см. в главе 18, “Проектирование системы”.

Регистр глобального управления асинхронной памятью (EBIU_AMGCTL)

В регистре глобального управления асинхронной памятью задаются различные аспекты контроллера. Он содержит биты активации банков и управления другими параметрами, описываемые в этом разделе. Не следует изменять значение этого регистра во время работы АМС. При задании конфигурации доступа процессора к внешним асинхронным устройствам, отображённым в карте памяти, запись в регистр EBIU_AMGCTL должна выполняться в последнюю очередь.

Регистр глобального управления асинхронной памятью (EBIU_AMGCTL)

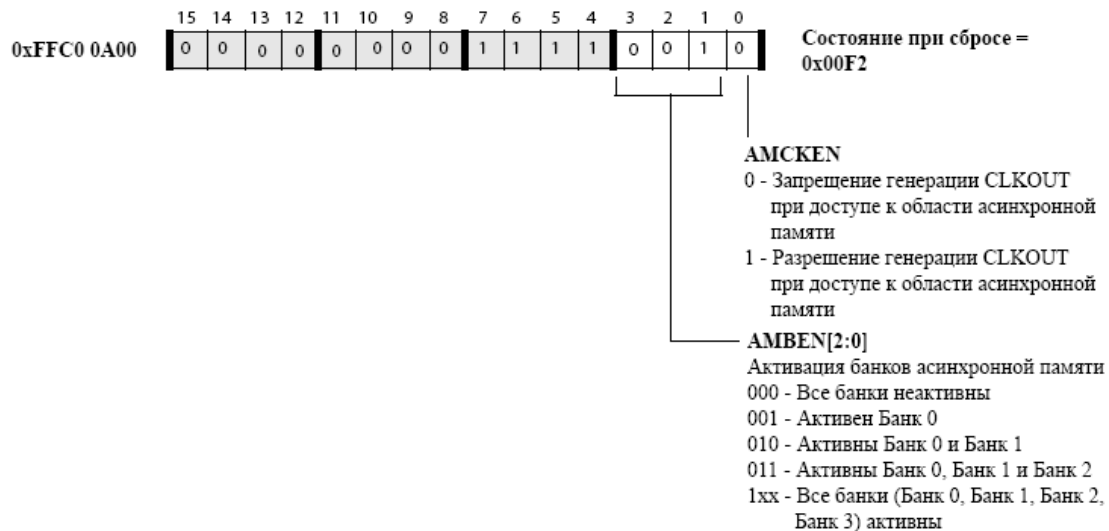


Рис. 17-3. Регистр управления асинхронной памятью

Если при выполнении операции шины производится обращение к неактивному банку асинхронной памяти, EBIU подтверждает передачу и устанавливает сигнал ошибки на шине, по которой поступает запрос. Сигнал ошибки передаётся ведущему устройству шины. Если инициатором запроса является ядро, генерируется аппаратное исключение ядра. Когда запрос выполняется ведущим DMA-совместимым устройством, ошибка регистрируется в соответствующем регистре состояния. Если банк не полностью заполнен памятью, то адреса незанятых областей банка являются псевдонимами (альтернативными адресами)

17 Интерфейс внешней шины

адресов заполненной области. EBIU не обнаруживает обращения к альтернативным адресам; сигнал ошибки в этом случае не устанавливается.

При работе с внешними устройствами, требующими тактового сигнала, можно разрешить генерацию CLKOUT, установив бит AMCKEN в регистре EBIU_AMGCTL. В системах, не использующих сигнал CLKOUT, бит AMCKEN следует установить в ноль.

Регистры управления банками асинхронной памяти (EBIU_AMBCTL0, EBIU_AMBCTL1)

Контроллер асинхронной памяти EBIU имеет два регистра управления банками памяти. Они содержат биты счётчиков времени установки, стробирования и удержания, биты настройки типа и размера памяти и биты параметров сигнала ARDY. Значение этого регистра не следует изменять во время работы АМС.

Временные характеристики АМС могут задаваться при помощи четырёх параметров:

- **Время установки:** время между началом цикла доступа к памяти (низким уровнем $\overline{AMS[x]}$) и установкой сигнала разрешения чтения (низким уровнем \overline{ARE}) или разрешения записи (низким уровнем \overline{AWE}).
- **Время доступа при чтении:** время между установкой и снятием сигнала разрешения чтения (низким и высоким уровнем \overline{ARE}).
- **Время доступа при записи:** время между установкой и снятием сигнала разрешения записи (низким и высоким уровнем \overline{AWE}).
- **Время удержания:** время между снятием сигнала разрешения чтения (высоким уровнем \overline{ARE}) или разрешения записи (высоким уровнем \overline{AWE}) и окончанием цикла доступа к памяти (высоким уровнем $\overline{AMS[x]}$).

Значение каждого из этих параметров может быть программно задано в тактах тактового сигнала EBIU. Эти параметры ограничены снизу следующими значениями:

- Время установки ≥ 1 такт
- Время доступа чтения ≥ 1 такт
- Время доступа записи ≥ 1 такт
- Время удержания ≥ 0 тактов

17 Интерфейс внешней шины

Регистр управления банками асинхронной памяти 0 (EBIU_AMBCTL0)

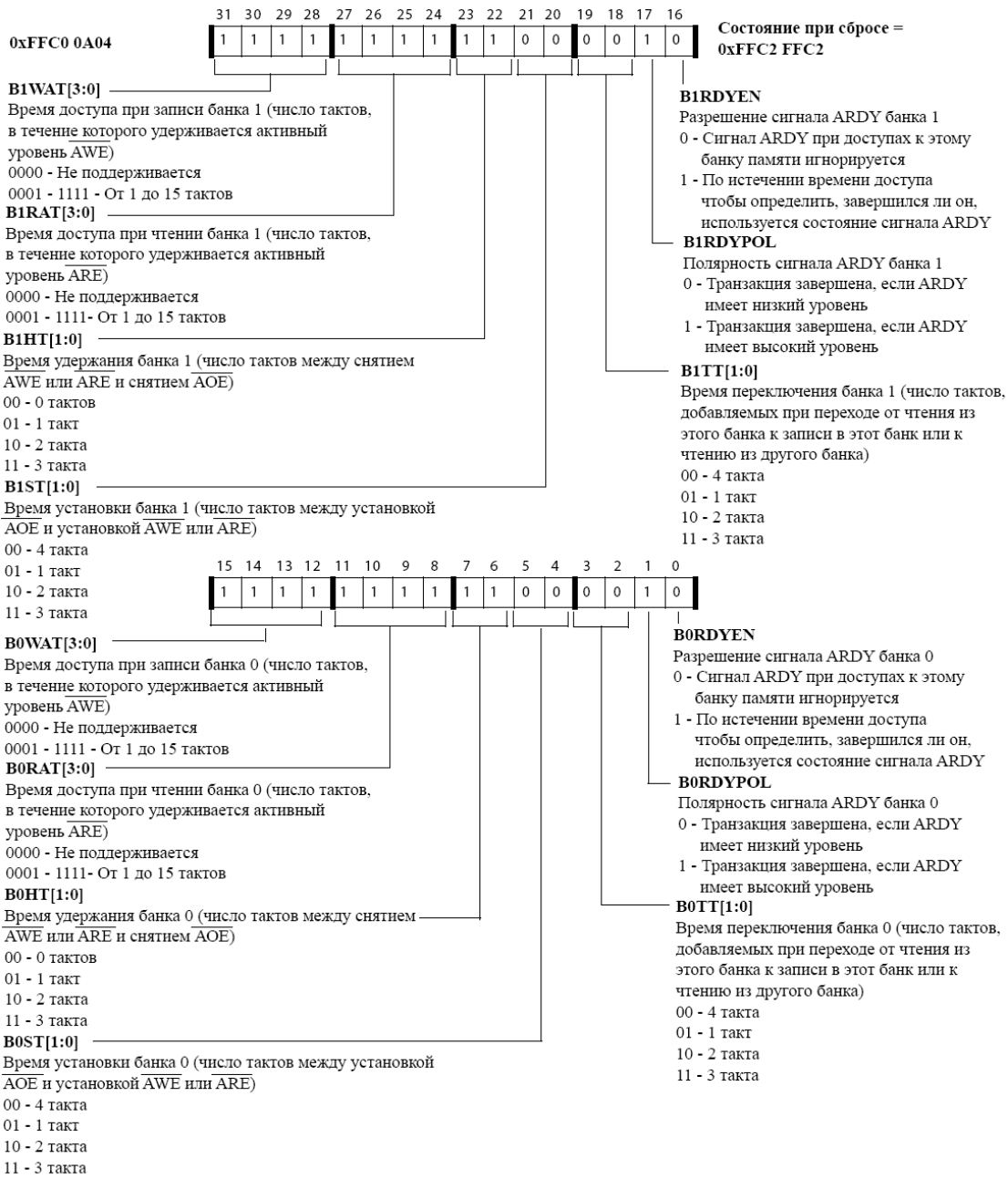


Рис. 17-4. Регистр управления банками асинхронной памяти 0

17 Интерфейс внешней шины

Регистр управления банками асинхронной памяти 1 (EBIU_AMBCTL1)

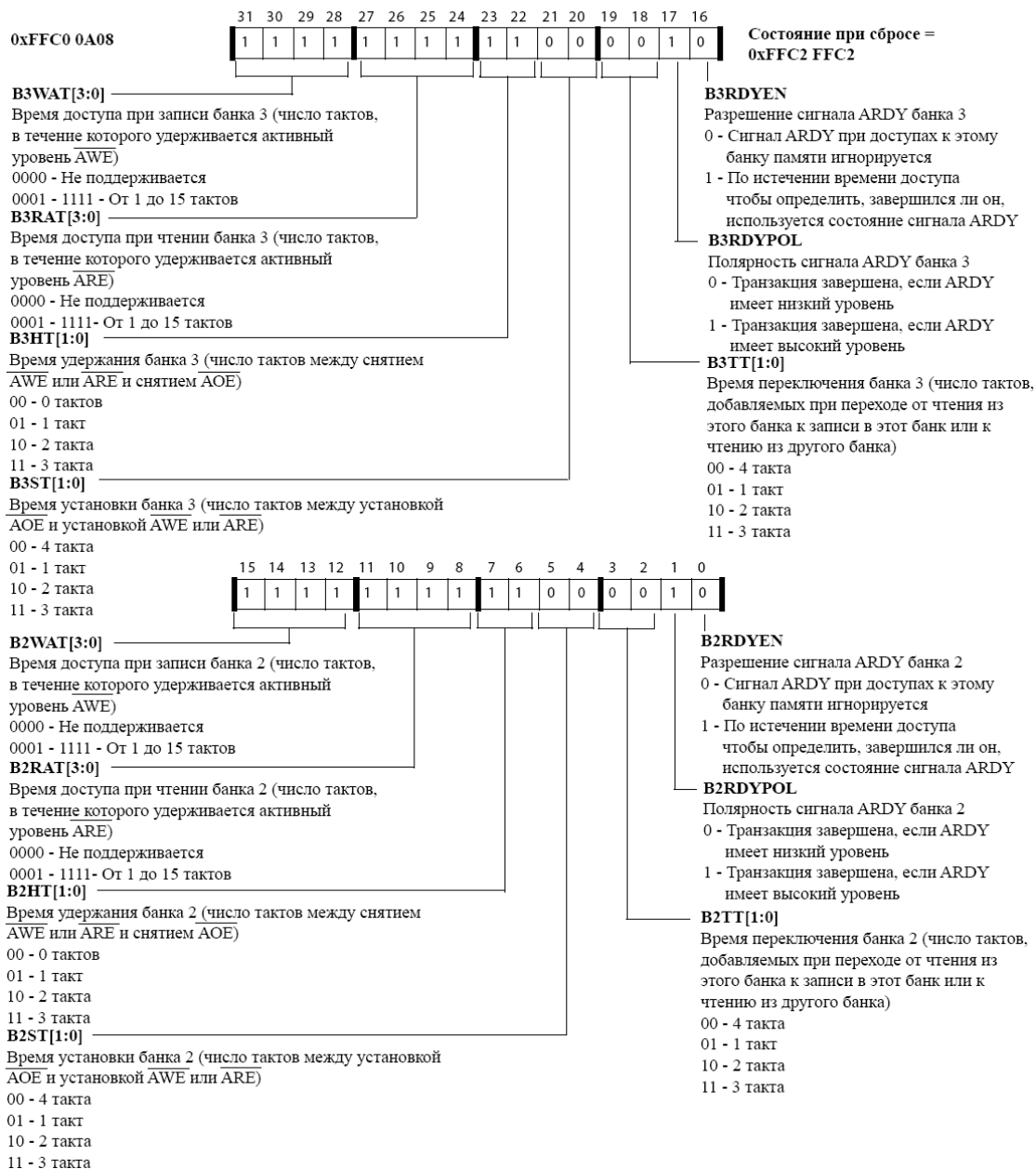


Рис. 17-5. Регистр управления банками асинхронной памяти 1

Избежание конфликтов на шине

Так как шина данных с тремя состояниями может совместно использоваться несколькими устройствами в системе, необходимо избегать конфликтных ситуаций на ней. Конфликтные ситуации вызывают чрезмерное рассеивание мощности и могут привести к сбою устройства. Они происходят в моменты времени, когда одно устройство освобождает, а другое захватывает шину (в случае если первое устройство переходит в третье состояние медленнее, чем второе начинает выдавать данные).

Конфликтная ситуация может возникнуть в следующих двух случаях:

17 Интерфейс внешней шины

- Чтение с последующей записью в одно пространство памяти. При этом формирователи шины данных могут конфликтовать с формирователями устройства памяти, адресуемого при чтении.
- Последовательное чтение из двух различных пространств памяти. При этом конфликт между двумя устройствами памяти, адресуемыми двумя операциями чтения, может возникать в момент перехода от одной операции к другой.

Для того чтобы избежать конфликтов, необходимо задать соответствующие значения времён изменения направления передачи (времён переключения банка) в регистрах управления банками асинхронной памяти. Эта опция позволяет задавать в программе для каждого банка число тактов тактового сигнала, добавляемое в указанных выше случаях. Минимальное количество тактов, добавляемое устройством интерфейса внешней шины (EBIU) при переключении, составляет один такт.

Управление входом ARDY

Для каждого банка программа может задавать необходимость опроса входа ARDY после достижения счётчиком чтения или записи нуля или игнорирование этого входного сигнала. Сигнал ARDY может использоваться для увеличения времени доступа. Необходимо отметить, что опрос ARDY выполняется синхронно, поэтому:

- Времена установки и снятия активного уровня сигнала ARDY должны соответствовать временам установки и удержания, указанным в спецификации процессора. Несоответствие указанным параметрам может привести к метастабильному поведению процессора. Для того чтобы гарантировать синхронность переходов сигнала ARDY следует использовать сигнал CLKOUT процессора.
- Сигнал на выводе ARDY должен находиться в стабильном состоянии (иметь либо низкий, либо высокий уровень) в такте, предшествующем обнулению внутреннего счётчика банка; то есть, в течение более одного такта CLKOUT перед ожидаемым передним фронтом \overline{AWE} или \overline{ARE} .
- Транзакция завершается в момент времени, когда при опросе ARDY имеет активный уровень.

Полярность сигнала ARDY может задаваться для каждого банка в отдельности. Опрос ARDY выполняется только при доступе к банку, для которого разрешено его использование; таким образом, по умолчанию подача сигнала ARDY не требуется. Дополнительную информацию см. в разделе “Добавление дополнительных состояний ожидания”.

Программируемые временные параметры

В этом разделе описываются программируемые временные параметры EBIU. Взаимосвязь временных параметров зависит от настройки АМС, инициатора транзакций (ядро или MemDMA) и последовательности транзакций (чтение с последующим чтением, чтение с последующей записью, и т.д.).

17 Интерфейс внешней шины

Доступ к асинхронным устройствам, инициируемый командами ядра

Некоторые типы обращений к внешней памяти вызываются командами ядра вида:

```
R0.L = W[R0++]; /* Чтение из внешней памяти. R0 указывает на ячейку внешней памяти */
```

или

```
W[R0++] = R0.L; /* Запись во внешнюю память */
```

Чтение из асинхронной памяти

На рис.17-6 показан цикл чтения из асинхронной памяти со следующими значениями временных параметров: время установки = 2 такта, время доступа при чтении = 2 такта, время удержания = 1 такт, время переключения = 1 такт.

В цикле чтения из асинхронной памяти выполняются следующие операции:

1. В начале периода установки выставляются достоверные сигналы шины адреса, $\overline{AMS}[x]$, и $\overline{ABE}[1 : 0]$, и устанавливается \overline{AOE} .
2. В начале периода доступа чтения, после двух тактов сигнала установки, выставляется сигнал \overline{ARE} .
3. В начале периода удержания, читаемые данные опрашиваются по переднему фронту тактового сигнала \overline{EBIU} . После этого фронта снимается активный уровень сигнала на выводе \overline{ARE} .
4. Если текущий цикл чтения не сопровождается другой операцией чтения из того же пространства асинхронной памяти, по окончании периода удержания снимается активный уровень сигнала \overline{AOE} .
5. Если текущая операция не сопровождается другой операцией чтения из того же банка памяти, АМС добавляет заданное число тактов времени переключения.

17 Интерфейс внешней шины

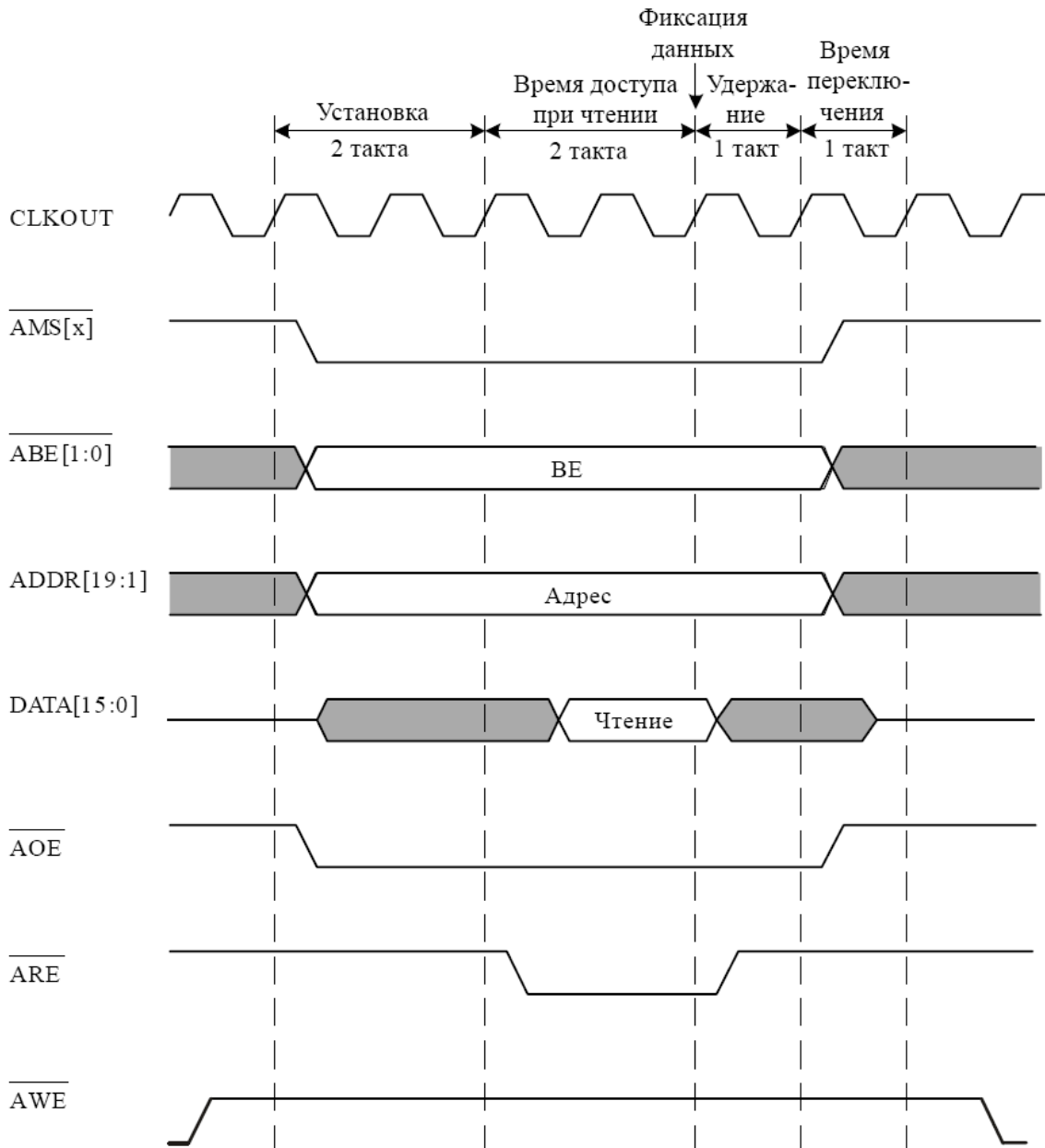


Рис. 17-6. Циклы чтения из асинхронной памяти

Запись в асинхронную память

На рис. 17-7 показан цикл записи в асинхронную память, сопровождаемый циклом чтения из того же банка, со следующими значениями временных параметров: время установки = 2 такта, время доступа при записи = 2 такта, время доступа при чтении = 3 такта, время удержания = 1 такт, время переключения = 1 такт.

В циклах записи в асинхронную память выполняются следующие операции:

1. В начале периода установки становятся истинными значения сигналов шины адреса, шины данных, $\overline{AMS[x]}$, и $\overline{ABE[1 : 0]}$.
2. В начале периода доступа записи выставляется сигнал \overline{AWE} .

17 Интерфейс внешней шины

3. В начале периода удержания снимается активный уровень сигнала \overline{AWE} .

В циклах чтения из асинхронной памяти выполняются следующие операции:

1. В начале периода установки становятся истинными значения сигналов шины адреса, $\overline{AMS[x]}$, и $\overline{ABE[1:0]}$, и выставляется сигнал \overline{AOE} .
2. В начале периода доступа чтения выставляется сигнал \overline{ARE} .
3. В начале периода удержания, читаемые данные опрашиваются по переднему фронту тактового сигнала EBIU. После этого фронта снимается активный уровень сигнала \overline{ARE} .
4. Если текущий цикл чтения не сопровождается другим циклом чтения из того же пространства асинхронной памяти, по окончании периода удержания снимается активный уровень сигнала \overline{AOE} . Если в следующем цикле производится доступ к другому банку памяти, также снимается активный уровень сигнала $\overline{AMS[x]}$.
5. Если текущая операция не сопровождается другой операцией чтения из того же банка памяти, АМС добавляет заданное число тактов времени переключения.

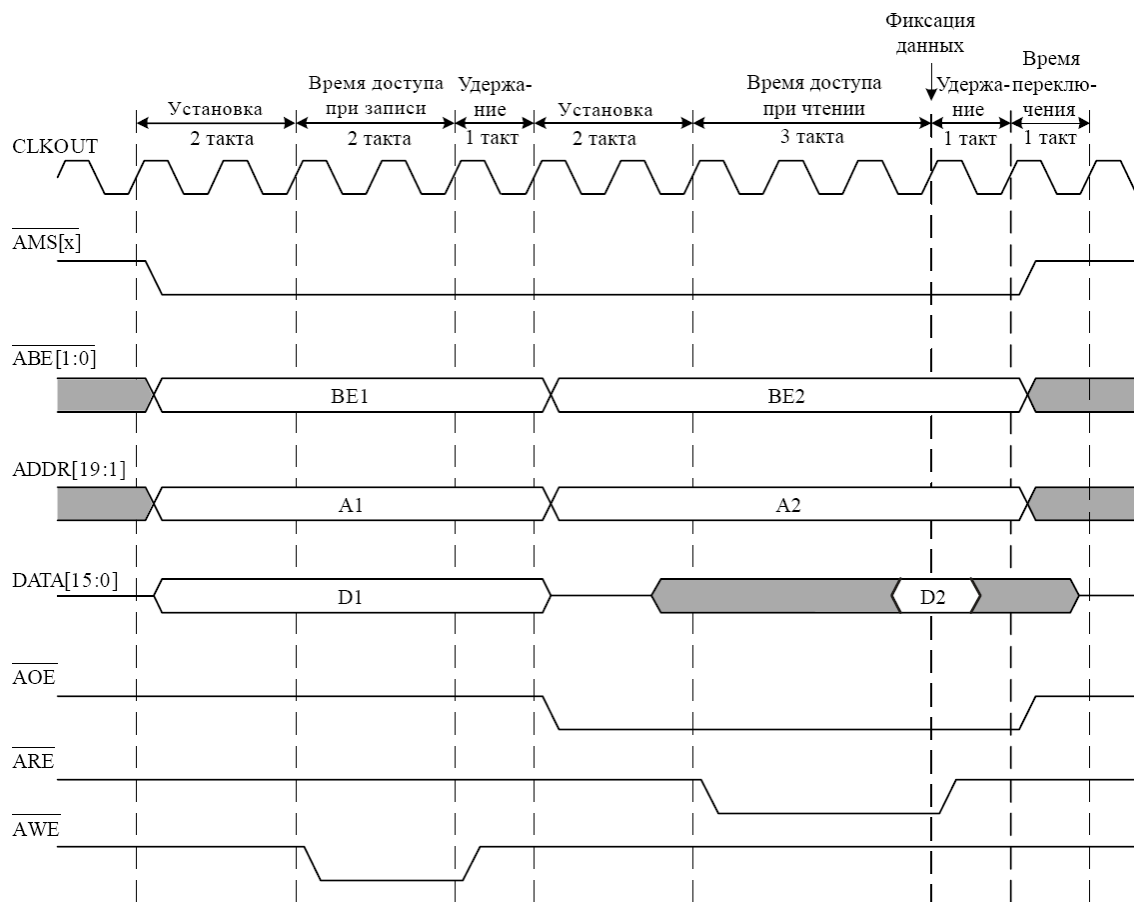


Рис. 17-7. Циклы записи в асинхронную память и чтения из неё

17 Интерфейс внешней шины

Добавление дополнительных состояний ожидания

Для добавления дополнительных состояний ожидания используется вывод ARDY. Входной сигнал на этом выводе опрашивается синхронно с тактовым сигналом EBIU. EBIU начинает опрос вывода ARDY за один такт тактового сигнала до конца заданного программно периода стробирования. Далее опрос выполняется по каждому последующему фронту тактового сигнала. Читаемые данные фиксируются по фронту тактового сигнала, на котором при опросе ARDY имеет высокий уровень. Пример временных диаграмм при использовании сигнала ARDY показан на рис. 17-8, где время установки = 2 такта, время доступа при чтении = 4 такта, время удержания = 1 такт. Необходимо отметить, что при использовании входа ARDY задаваемое время периода доступа при чтении должно составлять, по меньшей мере, два такта.

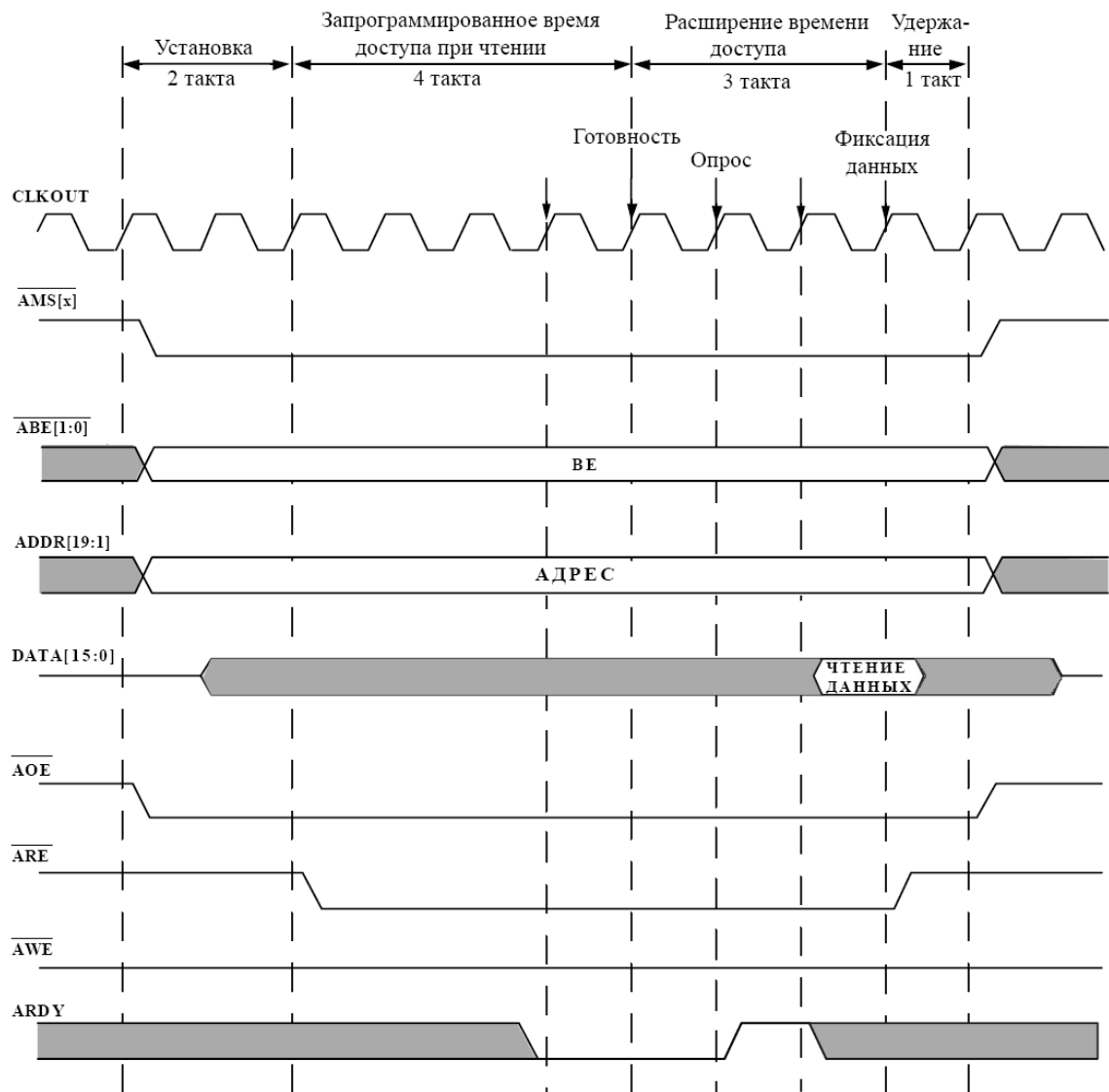


Рис. 17-8. Добавление состояний ожидания при помощи ARDY

17 Интерфейс внешней шины

Разрешение передачи байтов

При выполнении любых операций чтения из асинхронной памяти и 16-разрядных операций записи в асинхронную память на выводах $\overline{ABE}[1 : 0]$ выдаются сигналы высокого логического уровня. При выполнении записи в старший байт 16-разрядной асинхронной памяти $\overline{ABE1} = 0$ и $\overline{ABE0} = 1$. При записи в младший байт 16-разрядной асинхронной памяти $\overline{ABE1} = 1$ и $\overline{ABE0} = 0$.

Контроллер SDRAM (SDC)

Введение

Контроллер SDRAM (SDC) позволяет процессору обмениваться данными с синхронными DRAM (SDRAM) с максимальной частотой, определённой в *ADSP-BF531/ADSP-BF532/ADSP-BF533 Embedded Processor Data Sheet*. Процессор поддерживает интерфейс, не требующий внешних компонентов, с одним внешним банком SDRAM объёмом от 64 Мбит до 512 Мбит с организацией $\times 4$, $\times 8$ и $\times 16$ и максимальной полной ёмкостью 128 Мбайт. Этот банк управляется выводом выбора памяти SMS. Интерфейс позволяет задавать временные параметры для поддержки дополнительных буферов между процессором и SDRAM, обеспечивающих возможность работы с емкостной нагрузкой больших массивов памяти.

Все входы опрашиваются, и все выходные сигналы становятся истинными по переднему фронту выходного тактового сигнала SDRAM SCLKOUT.

SDC EBIU обеспечивает интерфейс со стандартными устройствами SDRAM, не требующий дополнительных компонентов. Контроллер SDRAM:

- поддерживает SDRAM объёмом 64 Мбит, 128 Мбит, 256 Мбит и 512 Мбит с организацией $\times 4$, $\times 8$ и $\times 16$;
- поддерживает до 128 Мбайт внешней SDRAM;
- поддерживает размеры страниц SDRAM – 512 байт, 1 Кбайт, 2 Кбайта и 4 Кбайта;
- поддерживает 4 внутренних банка в пределах области SDRAM;
- использует программируемый счётчик регенерации для координации между различными тактовыми частотами и требуемой частотой регенерации SDRAM;
- обеспечивает несколько опций временных параметров для поддержки дополнительных буферов между процессором и SDRAM;
- использует отдельный вывод (SA10), позволяющий выполнять предзаряд SDRAM до вызова команд авторегенерации или саморегенерации, в моменты времени, когда управление портом EBIU осуществляет контроллер асинхронной памяти;
- поддерживает саморегенерацию стандартных SDRAM и режим саморегенерации отдельных банков (partial array self-refresh) мобильных SDRAM;
- обеспечивает две опции включения SDRAM.

17 Интерфейс внешней шины

Терминология

В оставшейся части главы используются следующие термины.

Команда активации банка

По команде активации банка (Bank Activate) открывается строка (определяемая адресом строки) внутреннего банка (определяемого адресом банка) SDRAM. При вызове этой команды SDRAM открывает строку с новым адресом в назначенном банке. Память в открытой строке внутреннего банка называется открытой страницей. Команда активации банка должна применяться перед вызовом команды чтения или записи.

Длина пакета

Длина пакета определяет количество слов, сохраняемое в устройстве SDRAM или выдаваемое им при обнаружении отдельной команды записи или чтения, соответственно. Длина пакета задаётся записью определённых битов в регистр режима SDRAM во время выполнения процедуры включения SDRAM.

- ⓘ Несмотря на то, что при доступе к SDRAM в пакетном режиме SDC поддерживает только длины пакета, равные единице, он выполняет команды чтения и записи в каждом такте и продолжает обращение к данным. Установка значения длины пакета, равного единице, не снижает производительность.

Команда остановки передачи пакета

Вызов команды остановки передачи пакета (Burst Stop) является одним из способов прекратить или прервать операцию чтения или записи пакета.

- ⓘ Так как длина пакета SDRAM жёстко приравнена к единице, SDC не поддерживает команду остановки передачи пакета.

Тип пакета

Тип пакета определяет порядок адресов, в соответствии с которым SDRAM выдаёт или сохраняет пакет данных при обнаружении команд чтения и записи, соответственно. Тип пакета программируется в SDRAM во время выполнения процедуры включения.

- ⓘ Так как длина пакета SDRAM жёстко приравнена к единице, тип пакета не имеет значения. Однако SDC всегда устанавливает во время выполнения процедуры включения тип пакета “только с последовательным доступом” (sequential-accesses-only).

17 Интерфейс внешней шины

Задержка CAS (CL, CAS Latency)

Задержка между обнаружением устройством SDRAM команды чтения и выдачей данных на его выводы, выраженная в тактах тактового сигнала, называется задержкой строба адреса столбца (CAS, Column Address Strobe). Задержка CAS программируется в регистре режима SDRAM во время выполнения процедуры включения.

Величина задержки CAS определяется градацией скорости SDRAM и тактовой частотой проекта. SDC поддерживает значения задержки CAS, равные двум или трём тактам. Выбранное значение задержки CAS должно быть записано в регистр глобального управления SDRAM (EBIU_SDGCTL) до начала выполнения процедуры включения SDRAM. См. раздел “Регистр глобального управления SDRAM (EBIU_SDGCTL)”.

Цикл регенерации или саморегенерации CBR (CAS before RAS)

По истечении счётчика регенерации SDC выполняет предзаряд всех четырёх банков SDRAM и затем вызывает команду авторегенерации. По этой команде устройства SDRAM выполняют внутренний цикл регенерации CBR. По завершении внутренней регенерации выполняется предзаряд всех четырёх внутренних банков SDRAM.

Функция маскирования ввода/вывода данных DQM

Выводы SDQM[1:0] позволяют маскировать данные при выполнении 8- или 16-разрядных операций записи в SDRAM. Выводы DQM используются для блокирования выходного буфера SDRAM при выполнении предзаряда или определённых операций записи. Они не используются для маскирования данных в циклах чтения.

Внутренний банк

Отдельно взятое устройство SDRAM состоит из нескольких внутренних банков памяти. SDC поддерживает одиночные обращения к внутренним банкам. Адрес банка является частью адреса строки. SDC рассматривает все подключаемые устройства SDRAM, как имеющие четыре внутренних банка, а каждый активированный банк имеет уникальный адрес строки.

Регистр режима

Устройства SDRAM содержат внутренний регистр конфигурации, который позволяет задавать параметры их функционирования. Запись контроллера SDRAM в регистр режима должна вызываться программой в интервале между выполнением процедуры включения SDRAM и обращением (чтением/записью) к пространству SDRAM. Запись в регистр режима SDRAM вызывается записью

17 Интерфейс внешней шины

единицы в бит PSSE регистра глобального управления SDRAM (EBIU_SDGCTL) с последующим вызовом операции чтения из адресного пространства SDRAM или записи в него. Первая операция чтения или записи вызывает выполнение процедуры включения SDRAM, в ходе которой в регистр режима SDRAM копируется значение задержки CAS из регистра EBIU_SDGCTL. Для завершения первой операции чтения или записи требуется большое количество тактов. Следует отметить, что в большинстве проектов процедура включения SDRAM и запись в регистр режима выполняется только один раз. По завершении процедуры включения не следует повторно устанавливать бит PSSE, за исключением случаев, когда требуется изменение регистра режима.

Малопотребляющие устройства SDRAM также могут содержать расширенный регистр режима. EBIU позволяет производить программирование этого регистра во время выполнения процедуры включения при помощи бита EMREN регистра EBIU_SDGCTL.

Размер страницы

Количество памяти, диапазон адресов которой совпадает с диапазоном адресов строки. При последовательных обращениях к странице не требуется активация другой строки. Для 16-разрядных банков SDRAM размер строки определяется по следующей формуле:

$$\text{размер страницы} = 2^{(\text{CAW} + 1)},$$

где CAW – разрядность адреса столбца SDRAM; единица прибавляется, так как разрядность банка SDRAM равна 16 битам (1 бит адреса = 2 байта).

Команда предзаряда

По команде предзаряда (Precharge) закрывается определённый внутренний банк в активной странице или все внутренние банки в странице.

Банк SDRAM

Банк SDRAM это область памяти, выбираемая выводом SMS. Объём банка SDRAM может задаваться равным 16, 32, 64 или 128 Мбайт.



Не путайте “внутренние банки SDRAM”, являющиеся внутренними банками SDRAM и выбираемые по адресу банка, с “банком SDRAM” или “внешним банком”, активизируемым выводом SMS.

Саморегенерация

При работе SDRAM в режиме саморегенерации внутренний таймер SDRAM периодически инициирует циклы авторегенерации. При этом использование

17 Интерфейс внешней шины

внешнего управления не требуется. Для перевода SDRAM в режим низкого энергопотребления SDC должен вызвать последовательность команд, включая команду саморегенерации. Для выхода из режима саморегенерации контроллером SDRAM должна быть выполнена другая последовательность команд. Вход в режим саморегенерации программируется в регистре глобального управления SDRAM (EBIU_SDGCTL); любое обращение к адресному пространству SDRAM вызывает выход SDRAM из режима саморегенерации. См. раздел “Вход в режим саморегенерации (SRFS) и выход из него”.

t_{TRAS}

Требуемая задержка между командой активации банка и командой предзаряда, а также между командой саморегенерации и выходом из режима саморегенерации. Поле TRAS в регистре глобального управления SDRAM (EBIU_SDGCTL) содержит 4 бита и позволяет задавать значения задержки от 1 до 15 тактов. См. раздел “Выбор задержки команды активации банка (TRAS)”.

t_{TRP}

Требуемая задержка между вызовом команды предзаряда и

- вызовом команды активации банка,
- вызовом команды авторегенерации,
- вызовом команды саморегенерации.

Поле TRP в регистре глобального управления SDRAM (EBIU_SDGCTL) содержит 3 бита и позволяет задавать значения задержки от 1 до 7 тактов. См. раздел “Выбор задержки предзаряда (TRP)”.

t_{TRCD}

Требуемая задержка между командой активации банка и началом первой команды чтения или записи. Поле TRCD в регистре глобального управления SDRAM (EBIU_SDGCTL) содержит 3 бита и позволяет задавать значения задержки от 1 до 7 тактов.

t_{TRW}

Требуемая задержка между командой записи (выдачей записываемых данных) и командой предзаряда. Поле TRW в регистре глобального управления SDRAM (EBIU_SDGCTL) содержит 2 бита и позволяет задавать значения задержки от 1 до 3 тактов.

17 Интерфейс внешней шины

t_{RC}

Требуемая задержка между последовательным вызовом команд активации одного и того же внутреннего банка SDRAM. Эта задержка в явном виде не программируется. Значения полей TRAS и TPR должны задаваться с учётом t_{RC} таким образом, что $t_{RAS} + t_{RP} \geq t_{RC}$.

t_{RFC}

Требуемая задержка между вызовами команд авторегенерации и активации банка, и между последовательными вызовами команды авторегенерации. Эта задержка в явном виде не программируется и полагается равной t_{RC} . Значения полей TRAS и TPR должны задаваться с учётом t_{RC} таким образом, что $t_{RAS} + t_{RP} \geq t_{RC}$.

t_{RC}

Требуемая задержка между выходом из режима саморегенерации и вызовом команды авторегенерации. Эта задержка в явном виде не программируется и полагается равной t_{RC} . Значения полей TRAS и TPR должны задаваться с учётом t_{RC} таким образом, что $t_{RAS} + t_{RP} \geq t_{RC}$.

Поддерживаемые конфигурации SDRAM

В таблице 17-4 показаны все размеры, разрядности банков и конфигурации дискретных компонентов SDRAM, которые могут взаимодействовать с SDC без использования внешних компонентов.

Таблица 17-4. Поддерживаемые организации дискретных компонентов SDRAM

Размер банка (Мбайты)	Разрядность банка (биты)	SDRAM		
		Размер (Мбиты)	Конфигурация	Число кристаллов
16	16	32	2M × 4 × 4 банка	4
16	16	64	2M × 8 × 4 банка	2
16	16	128	2M × 16 × 4 банка	1
32	16	64	4M × 4 × 4 банка	4
32	16	128	4M × 8 × 4 банка	2
32	16	256	4M × 16 × 4 банка	1
64	16	128	8M × 4 × 4 банка	4
64	16	256	8M × 8 × 4 банка	2
64	16	512	8M × 16 × 4 банка	1
128	16	256	16M × 4 × 4 банка	4
128	16	512	16M × 8 × 4 банка	2
128	16	1024	16M × 16 × 4 банка	1

17 Интерфейс внешней шины

Пример блок-схемы системы с SDRAM

На рис. 17-9 показана блок-схема интерфейса SDRAM. В данном примере к интерфейс SDRAM подключается два 64-мегабитных ($\times 8$) устройства SDRAM и формируется один внешний банк памяти объемом 16 Мбайт. Устройства SDRAM используют общие шины адреса и управления.

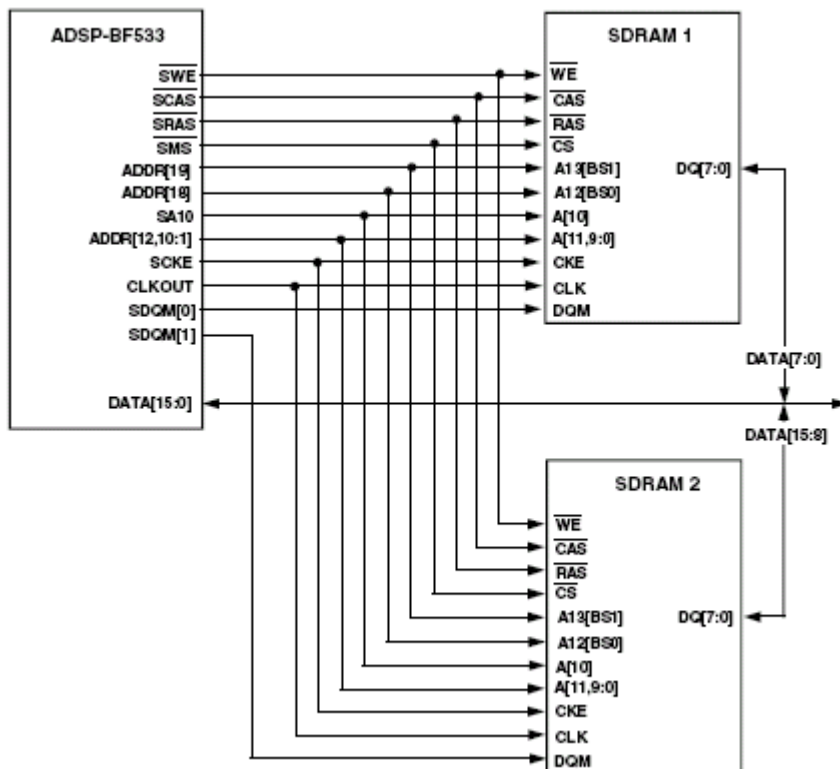


Рис. 17-9. Пример системы с SDRAM объемом 16 Мбайт

Регистр глобального управления SDRAM (EBIU_SDGCTL)

Регистр глобального управления SDRAM, показанный на рис. 17-10, содержит все программируемые параметры, связанные с конфигурацией и временами доступа к SDRAM.

17 Интерфейс внешней шины

Регистр глобального управления SDRAM (EBIU_SDGCTL)

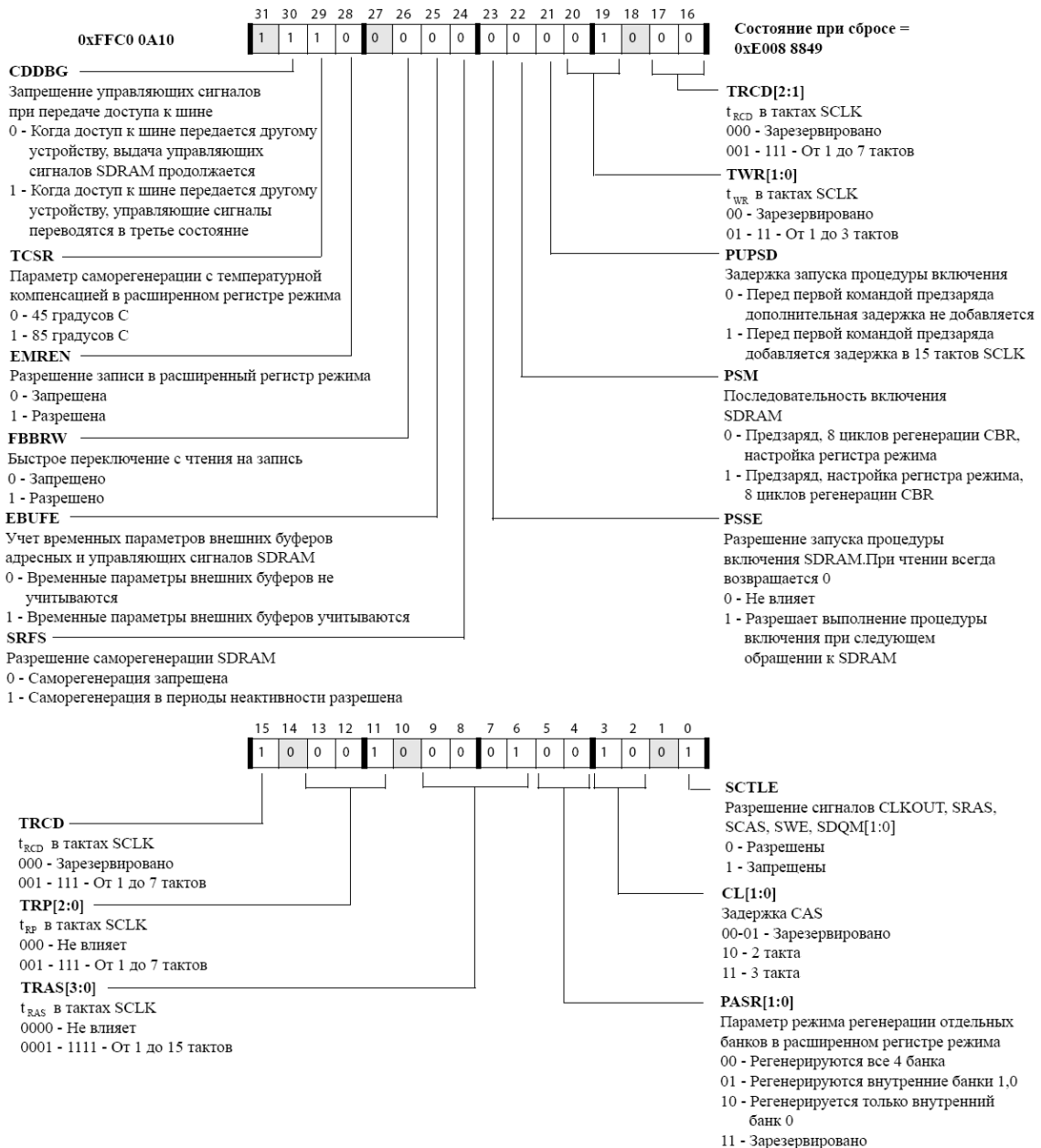


Рис. 17-10. Регистр глобального управления SDRAM

Бит SCTLE используется для разрешения или запрещения работы SDC. Если бит SCTLE сброшен, любое обращение к адресному пространству SDRAM генерирует ошибку внутренней шины. Обращение к внешней памяти при этом не производится. Дополнительную информацию см. в разделе “Обнаружение ошибок”. Когда SCTLE сброшен, все управляющие выходы SDC неактивны, и тактовый сигнал SDRAM не выдаётся. Для выполнения операций SDC бит SCTLE должен быть установлен. По умолчанию после сброса этот бит установлен.

Биты задержки CAS (CL) и времён t_{RAS} (TRAS), t_{RP} (TRP), t_{RCD} (TRCD) и t_{WR} (TWR) SDRAM необходимо программировать в соответствии со значением тактовой частоты системы и временными параметрами используемого устройства SDRAM.

17 Интерфейс внешней шины

Необходимо отметить, что все банки SDRAM имеют одинаковые временные параметры.

- ❗ Необходимо отметить, что указанные временные параметры не должны изменяться, когда SDC активен. По умолчанию после сброса работа SDC разрешена; таким образом, для модификации значений битов регистра `EBIU_SDGCTL`, установленных по умолчанию, работу SDC необходимо запретить.

Биты `PSM` и `PSSE` используются для задания режима и инициации процедуры включения (инициализации) SDRAM. Если `PSM = 1`, SDC выполняет команду полного предзаряда (`Precharge All`), команду загрузки регистра режима, после чего выполняет восемь циклов авторегенерации. Если `PSM = 0`, SDC выполняет команду полного предзаряда, затем восемь циклов авторегенерации, и затем команду загрузки регистра режима. Перед выполнением контроллером SDRAM процедуры включения SDRAM необходимо выполнить два действия:

- Для разрешения начала процедуры включения SDRAM необходимо установить бит `PSSE` в единицу.
- Для того чтобы SDC получил доступ к внешней шине, и стало возможным выполнение процедуры включения SDRAM, необходимо выполнить доступ (чтение или запись) к активному адресному пространству SDRAM.

Операция чтения или записи, используемая для запуска процедуры включения SDRAM, выполняется сразу после её завершения. То есть, первая операция чтения или записи в SDRAM выполняется с задержкой, так как процесс инициализации SDRAM занимает много тактов.

- ❗ Перед выполнением процедуры включения SDRAM необходимо убедиться, что на SDRAM поступает стабильное напряжение питания, и тактовый сигнал устройства SDRAM соответствует параметрам, указанным в его спецификации.

При использовании бита задержки начала процедуры включения (`PUPSD`) начало процедуры включения задерживается на 15 тактов `SCLK`. Это свойство является полезным в многопроцессорных системах, в которых используется общая SDRAM. Если до выполнения процедуры включения доступ к шине был предоставлен другому процессору, и во время передачи управления шиной SDRAM находилась в режиме саморегенерации, то для того, чтобы период ожидания от саморегенерации до первой команды предзаряда гарантированно превышал заданное время выхода из саморегенерации (t_{XSR}), можно использовать бит `PUPSD`.

Когда бит `SRFS` установлен, запускается режим саморегенерации. SDC выполняет последовательность команд, переводящую SDRAM в режим саморегенерации, сразу по завершении любой активной передачи. При следующем обращении к активному банку SDRAM SDC выполняет последовательность команд, выводящих SDRAM из режима саморегенерации, после чего выполняется запрашиваемая операция. Дополнительную информацию о бите `SRFS` см. в разделе “Вход в режим саморегенерации (`SRFS`) и выход из него”.

17 Интерфейс внешней шины

Бит `EBUFE` используется для разрешения/запрещения учёта временных параметров внешних буферов. При работе с буферизированными модулями SDRAM или при использовании для подачи сигналов на управляющие входы SDRAM дискретных буферных регистров, бит `EBUFE` следует установить в единицу. В этом случае при выполнении операций чтения или записи добавляются дополнительные такты на буферизацию данных. Дополнительную информацию о бите `EBUFE` см. в разделе “Настройка параметров буферизации SDRAM”.

Бит `FVBRW` разрешает выполнение записи в SDRAM на следующем такте после чтения. Во многих системах это невозможно, так как время, затрачиваемое на изменение направления выводов данных SDRAM очень велико, и при выполнении записи сразу после чтения, могут возникать конфликтные ситуации на шине. Если этот бит равен нулю, между операцией чтения и последующей операции записи добавляется один такт.

Бит `EMREN` разрешает программирование расширенного регистра режима при запуске. Расширенный регистр режима используется в некоторых малопотребляющих мобильных SDRAM для управления потреблением мощности. Когда бит `EMREN` установлен, биты `TCSR` и `PASR[1:0]` определяют значение, записываемое в регистр расширенного режима. Биты `PASR` задают количество внутренних банков SDRAM, регенерируемых в процессе саморегенерации. Бит `TCSR` задаёт наихудший температурный диапазон работы системы, и, таким образом, определяет, как часто необходимо выполнять регенерацию внутренних банков SDRAM в процессе саморегенерации.

Бит `CDDBG` используется для разрешения или запрещения управляющих сигналов SDRAM при передаче доступа к внешней памяти внешнему контроллеру. Если этот бит равен единице, при предоставлении доступа к шине другому устройству управляющие сигналы переводятся в третье состояние. В противном случае процессор продолжает выдавать управляющие сигналы. Когда этот бит установлен и доступ к шине передан другому устройству, предполагается, что внешний контроллер изменяет состояние всех внутренних банков SDRAM. Таким образом, когда SDC вновь получает доступ к внешней шине, перед началом выполнения операций необходимо произвести предзаряд SDRAM. Этот бит влияет на управляющие сигналы \overline{SRAS} , \overline{SCAS} , \overline{SWE} , `SMS`, `SA10`, `SCKE` и `CLKOUT`.

Необходимо отметить, что при записи в этот регистр во все зарезервированные биты должны записываться нули.

Разрешение тактового сигнала SDRAM (SCTLE)

Бит `SCTLE` позволяет программно блокировать все выходы управляющих сигналов SDRAM (`SDQM[3:0]`, \overline{SCAS} , \overline{SRAS} , \overline{SWE} , `SCKE` и `CLKOUT`).

- `SCTLE` = 0

17 Интерфейс внешней шины

Все управляющие выходы SDRAM блокируются (на управляющие выходы выдаются сигналы неактивного уровня, на вывод CLKOUT выдаётся сигнал низкого логического уровня).

- SCTL0 = 1
Разрешается выдача сигналов на всех управляющих выводах SDRAM (уровень сигнала на выводе CLKOUT изменяется).

Необходимо отметить, что CLKOUT используется совместно с AMC. Даже если бит SCTL0 сброшен, возможно независимое разрешение сигнала CLKOUT в AMC (при помощи бита AMCKEN в регистре EBIU_AMGCTL).

Если SDRAM в системе не используется, следует установить бит SCTL0 в ноль.

Если при SCTL0 = 0 выполняется обращение к адресному пространству SDRAM, генерируется ошибка внутренней шины и обращение к внешней памяти не производится. Дополнительную информацию см. в разделе “Обнаружение ошибок”. При соответствующем программном управлении бит SCTL0 может использоваться в режиме саморегенерации для дальнейшего снижения потребляемой мощности. Однако, при необходимости вызова контроллером SDRAM команд авторегенерации, бит SCTL0 должен постоянно оставаться в установленном состоянии.

Вход в режим саморегенерации и выход из него (SRFS)

SDC поддерживает режим саморегенерации SDRAM. В режиме саморегенерации устройство SDRAM выполняет операции регенерации внутренне (без внешнего управления); за счёт этого достигается снижение потребляемой им мощности.

Бит SRFS в регистре EBIU_SDGCTL разрешает запуск режима саморегенерации:

- SRFS = 0
Запрещение режима саморегенерации
- SRFS = 1
Разрешение режима саморегенерации

При SRFS = 1, когда SDC входит в состояние ожидания, он (при необходимости) вызывает команду предзаряда и, затем, команду саморегенерации. При внутреннем доступе к SDRAM SDC задерживает вызов команды саморегенерации до завершения выполняемой операции доступа к SDRAM и любых операций доступа, ожидающих выполнения. Дополнительную информацию см. в разделе “Команды SDC”.

При входе устройства SDRAM в режим саморегенерации контроллер SDRAM устанавливает бит SDSRA в регистре состояния управления SDRAM (EBIU_SDSTAT).

17 Интерфейс внешней шины

Устройство SDRAM выходит из режима саморегенерации только при получении контроллером SDRAM запроса доступа от ядра или контроллера DMA. Если бит SRFS был сброшен до получения запроса доступа, SDC остаётся в активном состоянии. Однако если бит SRFS по-прежнему установлен и отсутствуют операции обращения к памяти, которые необходимо выполнить, SDC снова входит в режим саморегенерации.

Необходимо отметить, что при $SRFS = 1$ по завершении обращений к памяти SDC входит в режим саморегенерации. Вход в режим саморегенерации не может быть отменён.

Настройка параметров буферизации SDRAM (EBUFE)

В системах с несколькими устройствами SDRAM, соединёнными параллельно, для удовлетворения требований к временным параметрам всей системы может потребоваться буферизация сигналов между процессором и устройствами SDRAM. Буфер обычно состоит из регистра и формирователя.

SDC может обеспечивать конвейеризацию адресных и управляющих сигналов SDRAM для удовлетворения требуемых временных параметров и поддержки промежуточной регистрации.

Этот режим разрешается битом EBUFE в регистре EBIU_SDGCTL:

- EBUFE = 0
Учёт временных параметров внешних буферов не производится
- EBUFE = 1
Временные параметры внешних буферов учитываются

Когда EBUFE = 1, контроллер SDRAM при записи задерживает данные на один такт для того, чтобы внешние буферные регистры зафиксировали адрес и внешние управляющие сигналы. При чтении контроллер SDRAM опрашивает данные на один такт позже, компенсируя задержку, вносимую внешними буферными регистрами. При разрешённой внешней буферизации задержка при любом обращении к SDRAM увеличивается на один такт.

Выбор величины задержки CAS (CL)

Величина задержки CAS определяет задержку между моментами обнаружения устройством SDRAM команды чтения и выдачи им данных на выходные выводы, выраженную в тактах тактового сигнала.

Задержка CAS не влияет на циклы записи.

Величина задержки CAS выбирается битами CL в регистре глобального управления SDRAM (EBIU_SDGCTL):

17 Интерфейс внешней шины

- CL = 00
Зарезервировано
- CL = 01
Зарезервировано
- CL = 10
Два такта
- CL = 11
Три такта

Величина задержки CAS определяется рабочей частотой. За более определённой информацией о настройке этого параметра обратитесь к документации на устройство SDRAM.

SDQM

Выходы SDQM[1:0] (Маска данных ввода/вывода) позволяет контроллеру SDRAM маскировать байты при байтовых передачах. Маскирование выполняется для каждого байта следующим образом: если сигнал на соответствующем выводе SDQM[x] имеет низкий логический уровень, запись данных разрешена, если SDQM[x] имеет высокий логический уровень, запись данных блокируется. В циклах записи операция маскирования данных не добавляет задержку.

Параллельное выполнение команды регенерации

SDC имеет отдельный адресный вывод (SA10), позволяющий выполнять команды авторегенерации параллельно с любым обращением к асинхронной памяти. Используя вывод SA10, SDC может вызывать команду предзаряда SDRAM до вызова команды регенерации. Этот вывод также позволяет SDC входить в режим самогенерации и выходить из него параллельно с любым обращением к асинхронной памяти.

Вывод SA10 должен быть подключён к выводу A10 SDRAM (а не к выводу ADDR[10]). Во время вызова команды предзаряда, SA10 используется в качестве индикатора необходимости выполнения полного предзаряда. Во время вызова команды активации банка на выводе SA10 выдаётся бит адреса внутренней строки, который должен быть мультиплексирован с входом A10 SDRAM. Во время вызова команд чтения и записи SA10 используется для блокирования функции автоматического предзаряда SDRAM.

17 Интерфейс внешней шины

Выбор задержки команды активации банка (TRAS)

Величина t_{RAS} (задержка команды активации банка) определяет требуемую задержку между вызовом контроллером SDRAM команды активации банка и команды предзаряда, выраженную в тактах тактового сигнала. Кроме того, SDRAM также должна оставаться в режиме саморегенерации, по меньшей мере, в течение периода времени, определяемого величиной t_{RAS} . Величины t_{RP} и t_{RAS} однозначно определяют значения t_{RFC} , t_{RC} и t_{XSR} .

Параметр t_{RAS} позволяет процессору адаптироваться к требуемым временным параметрам устройств SDRAM, используемым в системе.

Величина t_{RAS} задаётся битами TRAS в регистре глобального управления SDRAM (EBIU_SDGCTL). С их помощью может быть выбрано любое значение от 1 до 15 тактов. Например:

- TRAS = 0000
Не влияет на значение t_{RAS}
- TRAS = 0001
1 такт
- TRAS = 0010
2 такта
- TRAS = 1111
15 тактов

Информацию о настройке этого параметра для конкретного устройства SDRAM ищите в его документации.

Выбор задержки предзаряда (TRP)

Величина t_{RP} (задержка предзаряда) определяет требуемую задержку между моментом вызова контроллером SDRAM команды предзаряда и моментом вызова команды активации банка, выраженную в тактах тактового сигнала. Этот параметр также определяет требуемое время задержки между вызовом команд предзаряда и авторегенерации и между вызовом команд предзаряда и саморегенерации. Величины t_{RP} и t_{RAS} однозначно определяют значения t_{RFC} , t_{RC} и t_{XSR} .

Величина t_{RP} задаётся битами TRP в регистре глобального управления SDRAM (EBIU_SDGCTL). С их помощью может быть выбрано любое значение от 1 до 7 тактов. Например:

- TRP = 000
Не влияет на значение t_{RP}
- TRP = 001
1 такт

17 Интерфейс внешней шины

- TRP = 010
2 такта
- TRP = 111
7 тактов

Выбор задержки между записью и предзарядом (TWR)

Величина t_{WR} определяет требуемую задержку между моментом вызова контроллером SDRAM команды записи (выдачи записываемых данных) и моментом вызова команды предзаряда, выраженную в тактах тактового сигнала.

Этот параметр позволяет приложению адаптироваться к требуемым временным параметрам SDRAM.

Величина t_{WR} задаётся битами TWR в регистре глобального управления SDRAM (EBIU_SDGCTL). С их помощью может быть выбрано любое значение от 1 до 3 тактов. Например:

- TRP = 00
Зарезервировано
- TRP = 01
1 такт
- TRP = 10
2 такта
- TRP = 11
3 такта

Регистр управления банком памяти SDRAM (EBIU_SDBCTL)

Регистр управления банком памяти SDRAM содержит программируемые параметры внешнего банка и позволяет программе управлять некоторыми параметрами SDRAM. В нём могут задаваться различные размеры внешнего банка SDRAM. При доступе к внешнему банку используются временные параметры, определённые в регистре глобального управления SDRAM (EBIU_SDGCTL). Программирование регистра EBIU_SDBCTL следует выполнять до выполнения процедуры включения SDRAM. Значение этого регистра следует изменять только, когда SDC находится в состоянии ожидания.

17 Интерфейс внешней шины

Регистр управления банком SDRAM (EBIU_SDBCTL)

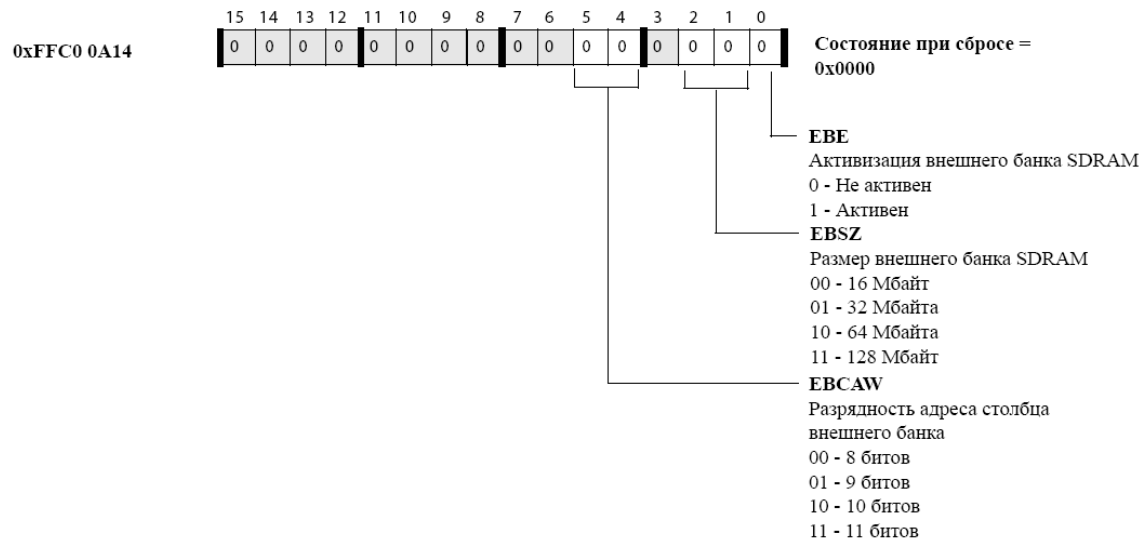


Рис. 17-11. Регистр управления банком памяти SDRAM

Регистр EBIU_SDBCTL хранит конфигурационную информацию интерфейса банка SDRAM. EBIU поддерживает устройства SDRAM объемом 64, 128, 256 и 512 Мбит с организацией $\times 4$, $\times 8$ и $\times 16$. В таблице 17-4 показаны значения объема SDRAM и разрядности шины данных, поддерживаемые при различных значениях поля EBSZ. Дополнительную информацию о дешифрации начального адреса банка см. в разделе “Размер внешней памяти SDRAM”.

SDC определяет размер внутренней страницы SDRAM на основании значения поля EBCAW. Поддерживаются размеры страниц 512 байт, 1 Кбайт, 2 Кбайта и 4 Кбайта. В таблице 17-5 показаны размеры страниц и разбиение внутреннего адреса (доступного ядру или контроллеру DMA как IA[31:0]) на адреса строки, банка, столбца и байта. Адрес внутри страницы определяется комбинацией адреса столбца и адреса байта.

Бит EBE в регистре EBIU_SDBCTL используется для разрешения или запрещения использования внешнего банка SDRAM. Если использование SDRAM запрещено, при любом обращении к адресному пространству SDRAM генерируется ошибка внутренней шины и обращение к внешней памяти не производится. Дополнительную информацию см. в разделе “Обнаружение ошибок”.

17 Интерфейс внешней шины

Таблица 17-5. Отображение внутреннего адреса

Разрядность банка (биты)	Размер банка (Мбайт)	Разрядность адреса столбца (CAW)	Размер страницы (Кбайт)	Адрес банка	Адрес строки	Страница	
						Адрес столбца	Адрес байта
16	128	11	4	IA[26:25]	IA[24:12]	A[11:1]	IA[0]
16	128	10	2	IA[26:25]	IA[24:11]	IA[10:1]	IA[0]
16	128	9	1	IA[26:25]	IA[24:10]	IA[9:1]	IA[0]
16	128	8	0.5	IA[26:25]	IA[24:9]	IA[8:1]	IA[0]
16	64	11	4	IA[25:24]	IA[23:12]	IA[11:1]	IA[0]
16	64	10	2	IA[25:24]	IA[23:11]	IA[10:1]	IA[0]
16	64	9	1	IA[25:24]	IA[23:10]	IA[9:1]	IA[0]
16	64	8	0.5	IA[25:24]	IA[23:9]	IA[8:1]	IA[0]
16	32	11	4	IA[24:23]	IA[22:12]	IA[11:1]	IA[0]
16	32	10	2	IA[24:23]	IA[22:11]	IA[10:1]	IA[0]
16	32	9	1	IA[24:23]	IA[22:10]	IA[9:1]	IA[0]
16	32	8	0.5	IA[24:23]	IA[22:9]	IA[8:1]	IA[0]
16	16	11	4	IA[23:22]	IA[21:12]	IA[11:1]	IA[0]
16	16	10	2	IA[23:22]	IA[21:11]	IA[10:1]	IA[0]
16	16	9	1	IA[23:22]	IA[21:10]	IA[9:1]	IA[0]
16	16	8	0.5	IA[23:22]	IA[21:9]	IA[8:1]	IA[0]

Регистр состояния управления SDRAM (EBIU_SDSTAT)

Регистр состояния управления SDRAM обеспечивает информацию о состоянии SDC. Эта информация может использоваться для определения того момента, когда существует возможность безопасного изменения параметров управления SDC, или для отладки. Бит SDEASE этого регистра является залипающим. Он сбрасывается программно записью единицы. Остальные биты обновляются только контроллером SDRAM, программная запись не влияет на их содержимое. Этот регистр SDC является 16-разрядным.

17 Интерфейс внешней шины

Регистр состояния управления SDRAM (EBIU_SDSTAT)

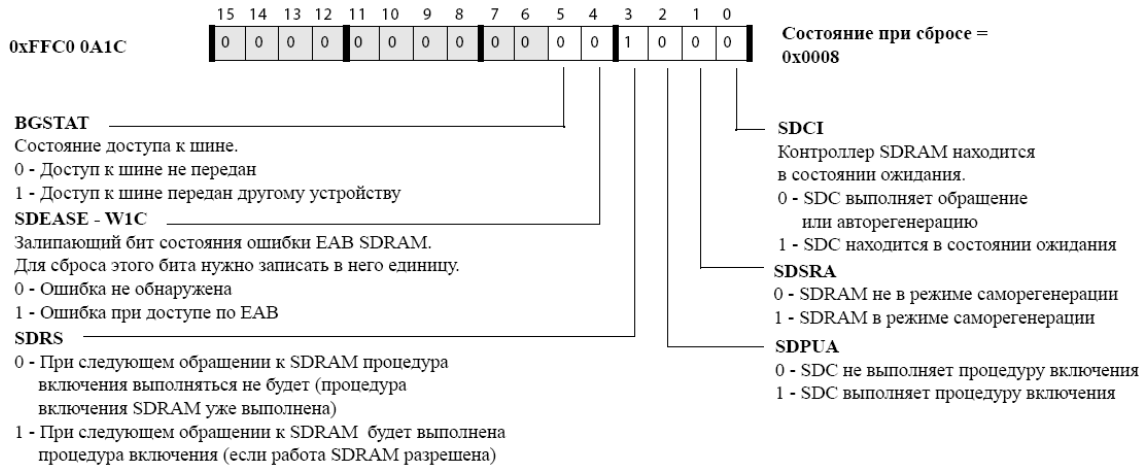


Рис. 17-12. Регистр состояния управления SDRAM

Регистр управления частотой регенерации SDRAM (EBIU_SDRRC)

Регистр управления частотой регенерации SDRAM обеспечивает гибкий механизм задания параметров авторегенерации. Так как частота тактового сигнала, подаваемого на SDRAM, может варьироваться, SDC содержит программируемый счётчик регенерации, период которого зависит от значения, запрограммированного в поле RDIV регистра EBIU_SDRRC. Этот счётчик координирует частоту подаваемого на SDRAM тактового сигнала с требуемой частотой регенерации.

В поле RDIV должно записываться требуемое значение периода счётчика регенерации (в тактах тактового сигнала SDRAM). Таймаут счётчика регенерации вызывает команду авторегенерации всех внешних устройств SDRAM. Запись значения RDIV в регистр EBIU_SDRRC необходимо выполнить до запуска процедуры включения SDRAM. Изменение значения RDIV возможно только, когда SDC находится в состоянии ожидания.

Регистр управления частотой регенерации SDRAM (EBIU_SDRRC)

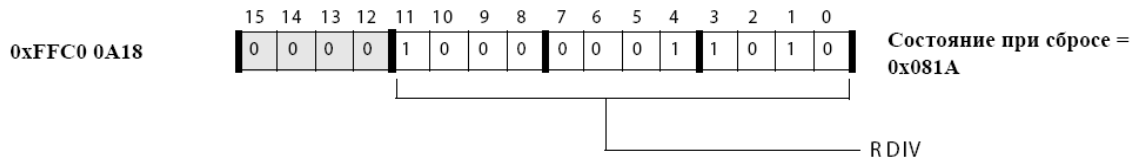


Рис. 17-13. Регистр управления частотой регенерации SDRAM

Для вычисления значения, которое следует записать в регистр EBIU_SDRRC, используйте следующую формулу:

17 Интерфейс внешней шины

$$RDIV = ((f_{SCLK} \times t_{REF}) / NRA) - (t_{RAS} + t_{RP}),$$

где:

- f_{SCLK} – тактовая частота SDRAM (тактовая частота системы),
- t_{REF} – период регенерации SDRAM,
- NRA – количество адресов строк SDRAM (количество циклов регенерации, необходимых для регенерации всего устройства SDRAM),
- t_{RAS} – задержка от вызова команды активации банка до вызова команды предзаряда (t_{RAS} в регистре глобального управления SDRAM), выраженная в тактах тактового сигнала,
- t_{RP} – задержка от RAS до вызова команды предзаряда (t_{RP} в регистре глобального управления SDRAM), выраженная в тактах тактового сигнала.

В этом выражении из числа тактов между циклами регенерации вычитается необходимая задержка между вызовами команд активации одного и того же внутреннего банка ($t_{RC} = t_{RAS} + t_{RP}$). Таким образом, если таймаут счётчика регенерации происходит, когда активен цикл SDRAM, то гарантированно будут удовлетворены требования к частоте регенерации SDRAM. Получаемый результат всегда должен округляться до меньшего целого.

Ниже приводится пример вычисления RDIV для типичного устройства SDRAM, работающего в системе с тактовой частотой 133 МГц:

$$\begin{aligned} f_{SCLK} &= 133 \text{ МГц} \\ t_{REF} &= 64 \text{ нс} \\ NRA &= 4096 \text{ адресов строк} \\ t_{RAS} &= 2 \\ t_{RP} &= 2 \end{aligned}$$

Выражение для вычисления RDIV принимает вид:

$$RDIV = ((133 \times 10^6 \times 64 \times 10^{-9}) / 4096) - (2 + 2) = 2074 \text{ такта тактового сигнала}$$

Таким образом, $RDIV = 0x81A$ (в шестнадцатеричной записи), и в регистр управления частотой регенерации SDRAM следует записать $0x081A$.

Необходимо отметить что, если работа контроллера SDRAM разрешена, в RDIV должно быть записано ненулевое значение. Работа контроллера SDRAM при $RDIV = 0$ не поддерживается; запись нулевого значения в RDIV может привести к нежелательным результатам. RDIV может принимать любые значения в диапазоне от $0x001$ до $0xFFFF$.

Размер внешней памяти SDRAM

Общее количество внешней памяти SDRAM, адресуемой процессором, определяется битами EBSZ регистра EBIU_SDBCTL (см. таблицу 17-6). Обращение к памяти, адрес которой выходит за границы диапазона, указанного для конкретного значения EBSZ, вызывает ошибку внутренней шины; обращение

17 Интерфейс внешней шины

Мультиплексирование адреса 16-разрядной SDRAM

В таблице 17-7 перечислены выводы устройства SDRAM, подключаемые к адресным выводам процессора.

Таблица 17-7. Подключение адресных выводов SDRAM для 16-разрядных банков

Внешний адресный вывод	Адресный вывод SDRAM
ADDR[19]	BA[1]
ADDR[18]	BA[0]
ADDR[16]	A[15]
ADDR[15]	A[14]
ADDR[14]	A[13]
ADDR[13]	A[12]
ADDR[12]	A[11]
SA[10]	A[10]
ADDR[10]	A[9]
ADDR[9]	A[8]
ADDR[8]	A[7]
ADDR[7]	A[6]
ADDR[6]	A[5]
ADDR[5]	A[4]
ADDR[4]	A[3]
ADDR[3]	A[2]
ADDR[2]	A[1]
ADDR[1]	A[0]

Маска данных (SDQM[1:0])

При записи в SDRAM выводы SDQM[1:0] используются для маскирования записи отдельных байтов. В таблице 17-8 показаны значения SDQM[1:0] для 16-разрядной SDRAM в зависимости от бита внутреннего адреса передачи IA[0] и размера передачи.

При чтении данных из банков SDRAM независимо от размера передачи всегда выполняется чтение всех байтов банка. Таким образом, при чтении из 16-разрядных банков SDRAM SDQM[1:0] всегда равны нулю.

Сигналы на выводах SDQM[1:0] имеют высокий уровень только во время маскирования данных при записи в SDRAM. Во всех остальных случаях сигналы на выводах SDQM[1:0] имеют низкий уровень.

Таблица 17-8. Значения сигналов на выводах SDQM[1:0] при записи

Внутренний адрес IA[0]	Внутренний размер передачи		
	байт	2 байта	4 байта
0	SDQM[1] = 1 SDQM[0] = 0	SDQM[1] = 0 SDQM[0] = 0	SDQM[1] = 0 SDQM[0] = 0
1	SDQM[1] = 0 SDQM[0] = 1		

17 Интерфейс внешней шины

Работа SDC

При чтении и записи контроллером SDRAM используется единичная длина пакета. При каждом промахе при обращении к странице SDC перед выполнением команды чтения или записи выполняет последовательность команд предзаряда и активации банка. При совпадении при обращении к странице, выполнение команды предзаряда не требуется, и SDC незамедлительно выполняет команду чтения или записи.

При выполнении чтения из SDRAM существует задержка от вызова команды чтения до доступности данных SDRAM (задержка CAS). Эта задержка всегда присутствует при первой операции чтения в пакетном режиме и при любом однократном чтении. Последующие операции чтения в пакетном режиме выполняются без задержки.

SDC содержит программируемый счётчик регенерации. Он может программироваться на генерацию циклов авторегенерации с требуемой частотой в фоновом режиме в соответствии с тактовой частотой SDRAM. Период счётчика регенерации задаётся полем RDIV в регистре управления частотой регенерации SDRAM.

Процессор имеет отдельный вывод A10 (SA10), позволяющий выполнять команды предзаряда параллельно с любыми операциями АМС. До вызова команды авторегенерации выполняется предзаряд всех внутренних банков SDRAM.

Внутренний немultipлексированный 32-разрядный адрес multiplexируется в адрес строки, адрес столбца, адрес выбора банка и маску данных. Для генерации масок данных в 16-разрядных устройствах SDRAM используется бит 0 адреса. Последующие биты формируют адрес строки и адрес столбца. Последние два бита формируют адрес внутреннего банка. Отображение адресов выполняется в зависимости от значений параметров EBSZ и EBCAW в регистре управления банком памяти SDRAM.

Конфигурирование SDC

После аппаратного или программного сброса генерация тактового сигнала SDC разрешена; однако SDC должен быть сконфигурирован и инициализирован. До программирования SDC и выполнения процедуры включения необходимо убедиться в том, что напряжение питания стабильно в течение необходимого времени (задаваемого SDRAM), после чего разрешить генерацию тактового сигнала SDRAM. Для того чтобы настроить SDC и запустить процедуру включения SDRAM необходимо произвести запись в регистр управления частотой регенерации SDRAM (EBIU_SDRRC), регистр управления банком памяти SDRAM (EBIU_SDBCTL) и регистр глобального управления SDRAM (EBIU_SDGCTL) и выполнить обращение к адресному пространству SDRAM. Текущее состояние SDC может быть определено при помощи бита SDRS регистра состояния

17 Интерфейс внешней шины

управления SDRAM. Если этот бит установлен, процедура включения SDRAM не инициировалась.

Для задания частоты регенерации SDRAM следует выполнить запись в поле RDIV регистра EBIU_SDRRC.

Значение, записываемое в регистр EBIU_SDBCTL, определяет размер и организацию используемого устройства памяти SDRAM (поля EBSZ и EBCAW) и разрешает/запрещает использование внешнего банка (EBE). Необходимо отметить что, независимо от состояния бита EBE, любое обращение к адресному пространству SDRAM до запуска процедуры включения SDRAM вызывает ошибку внутренней шины; при этом обращение к внешней памяти не производится. Дополнительную информацию см. в разделе “Обнаружение ошибок”. Если использование внешнего банка запрещено, по завершении процедуры включения SDRAM любое обращение к нему вызывает прерывание аппаратной ошибки; при этом обмен данными с SDRAM не выполняется.

Запись в регистр EBIU_SDGCTL выполняется в следующих целях:

- для настройки временных параметров циклов SDRAM (CL, TRAS, TRP, TRCD, TWR, EBUFE);
- для разрешения подачи тактового сигнала на SDRAM (SCTLE);
- для выбора и разрешения запуска процедуры включения SDRAM (PSM, PSSE).

Необходимо отметить что, если бит SCTLE сброшен, любое обращение к адресному пространству SDRAM вызывает ошибку внутренней шины; при этом обращение к внешней памяти не выполняется. Дополнительную информацию см. в разделе “Обнаружение ошибок”.

Когда бит PSSE регистра EBIU_SDGCTL равен единице, после обращения к разрешённому адресному пространству SDRAM SDC инициирует процедуру включения SDRAM. Порядок её выполнения определяется битом PSM регистра EBIU_SDGCTL. Операцией, используемой для запуска процедуры включения, может быть либо операция записи, либо операция чтения. Она завершается по завершении процедуры включения SDRAM. Для завершения первой операции может потребоваться большое количество тактов.

Команды SDC

В этом разделе описываются команды, используемые контроллером SDRAM для управления интерфейсом SDRAM. Эти команды автоматически инициируются при чтении из памяти или записи в память. Ниже приводится обзор различных команд, используемых внутрикристальным контроллером для управления интерфейсом SDRAM.

- Предзаряд: Выполняется предзаряд всех внутренних банков.
- Активация банка: Активируется страница в требуемом внутреннем банке SDRAM.

17 Интерфейс внешней шины

- Загрузка регистра режима: Во время выполнения процедуры включения инициализируются рабочие параметры SDRAM.
- Чтение/Запись.
- Авторегенерация: Вызывает выполнение устройством SDRAM цикла регенерации “CAS before RAS” (CBR).
- Саморегенерация: SDRAM переводится в режим саморегенерации (питание SDRAM отключается и операция регенерации управляется внутренне).
- NOP/Запрет выполнения команд(Command Inhibit): Операции не выполняются.

В таблице 17-9 показаны состояния выводов SDRAM при выполнении команд SDC.

Таблица 17-9. Уровни сигналов на выводах SDRAM при выполнении команд SDC

Команда	SMS	SCAS	SRAS	SWE	SCKE	SA10
Полный предзаряд	низкий	высокий	низкий	низкий	высокий	высокий
Активация банка	низкий	высокий	низкий	высокий	высокий	
Загрузка регистра режима	низкий	низкий	низкий	низкий	высокий	
Чтение	низкий	низкий	высокий	высокий	высокий	низкий
Запись	низкий	низкий	высокий	низкий	высокий	низкий
Авторегенерация	низкий	низкий	низкий	высокий	высокий	
Саморегенерация	низкий	низкий	низкий	высокий	низкий	
NOP	низкий	высокий	высокий	высокий	высокий	
Запрет выполнения команд	высокий	высокий	высокий	высокий	высокий	

Команда предзаряда

Команда предзаряда вызывается при необходимости предзаряда активного внутреннего банка. SDC всегда вызывает команду предзаряда для всех внутренних банков SDRAM (управляется установкой высокого уровня сигнала на выводе SA10). При промахе при обращении к странице (записи или чтении) предзаряжается только тот внутренний банк, к которому производится обращение.

Команда активации банка

Выполнение команды активации банка требуется, когда обращение производится к новой странице. Для активации страницы в требуемом внутреннем банке SDRAM SDC последовательно вызывает команды предзаряда и активации банка.

Команда загрузки регистра режима

Команда загрузки регистра режима инициализирует рабочие параметры SDRAM. Эта команда является частью процедуры включения SDRAM. Команда загрузки регистра режима использует адресную шину SDRAM как вход данных. Процедура включения иницируется записью единицы в бит PSSE регистра глобального управления SDRAM (EBIU_SDGCTL) с последующей записью или чтением по любому доступному адресу в адресном пространстве SDRAM. Точный порядок

17 Интерфейс внешней шины

выполнения процедуры включения определяется битом PSM регистра EBIU_SDGCTL.

Команда загрузки регистра режима инициализирует следующие параметры:

- длина пакета = 1, биты 2 – 0, всегда 0;
- порядок адресов в пакете = последовательный, бит 3, всегда 0;
- Ltmode = режим задержки (задержка CAS), биты 6-4, программируются в регистре EBIU_SDGCTL;
- биты 14 – 7, всегда 0.

При выполнении команды загрузки регистра режима сигналы на незадействованных адресных выводах всегда устанавливаются в 0. В следующих двух тактах после команды загрузки регистра режима SDC вызывает только команды NOP.

Если установлен бит EMREN в регистре EBIU_SDGCTL, при выполнении процедуры включения также может программироваться расширенный регистр режима малопотребляющих мобильных SDRAM.

Расширенный регистр режима инициализируется следующими параметрами:

- саморегенерация отдельных банков (partial array self-refresh), биты 2 – 0, бит 2 всегда 0, биты 1 – 0 программируются в регистре EBIU_SDGCTL;
- саморегенерация с температурной компенсацией, биты 4 – 3, бит 3 всегда 1, бит 4 программируется в регистре EBIU_SDGCTL;
- биты 12 – 5, всегда 0;
- бит 13 всегда 1.

Команда чтения/записи

Команда чтения/записи выполняется, если следующий доступ чтения/записи производится к текущей активной странице. По команде чтения SDRAM фиксирует адрес столбца. Задержка между командами активации и чтения определяется параметром t_{RCD} . Данные на выходе SDRAM становятся доступны после выдержки времени задержки CAS.

По команде записи SDRAM фиксирует адрес столбца. Достоверные записываемые данные появляются на том же такте. Задержка между командами активации и записи определяется параметром t_{RCD} .

SDC не использует функцию автоматического предзаряда SDRAM, активизируемую установкой высокого уровня сигнала на выводе SA10 во время выполнения команды чтения или записи.

Команда авторегенерации

По команде авторегенерации SDRAM внутренне инкрементирует счётчик адреса регенерации и вызывает для данного адреса цикл регенерации “CAS before RAS”

17 Интерфейс внешней шины

(CBR). Команда авторегенерации даётся контроллером SDRAM после таймаута счётчика регенерации SDC. Для того чтобы в течение периода t_{REF} , определяемого спецификациями на временные параметры SDRAM, была выполнена регенерация всех адресов, необходимо соответствующим образом запрограммировать поле RDIV в регистре управления частотой регенерации SDRAM. Эта команда вызывается для внешнего банка, независимо от того, активен он или нет (независимо от значения бита EBE в регистре глобального управления SDRAM). Перед выполнением команды авторегенерации SDC выполняет команду полного предзаряда внешнего банка. Следующая команда активации даётся только после выдержки временного интервала, определяемого параметром t_{RFC} ($t_{RFC} = t_{RAS} + t_{RP}$).

Команды авторегенерации также вызываются контроллером SDC при выполнении процедуры включения и после выхода из режима саморегенерации.

Команда саморегенерации

По команде саморегенерации SDRAM выполняет операции регенерации внутренне, без внешнего управления. То есть, когда SDRAM находится в режиме саморегенерации, SDC не генерирует циклы авторегенерации. Перед выполнением команды саморегенерации выполняется предзаряд всех внутренних банков. Режим саморегенерации разрешается записью единицы в бит SRF5 регистра глобального управления SDRAM (EBIU_SDGCTL). После вызова команды саморегенерации SDC выдаёт на вывод SCKE сигнал низкого уровня, что переводит SDRAM в режим пониженного энергопотребления ($SCKE = 0$, $\overline{SRAS} / \overline{SMS} / \overline{SCAS} / \overline{SWE} = 1$). Перед выходом из режима саморегенерации SDC устанавливает активный уровень сигнала SCKE. SDRAM остаётся в режиме саморегенерации в течение, по меньшей мере, t_{RAS} , пока не произойдёт внутреннее обращение к пространству SDRAM. При возникновении внутреннего обращения к пространству SDRAM, по которому производится вывод SDRAM из режима саморегенерации, SDC ожидает в течение t_{XSR} ($t_{XSR} = t_{RAS} + t_{RP}$), после чего вызывает команду авторегенерации. После выполнения команды авторегенерации SDC ожидает в течение t_{RFC} ($t_{RFC} = t_{RAS} + t_{RP}$) перед выполнением активации банка, обращение к которому вызвало вывод SDRAM из режима саморегенерации. Таким образом, при работе в режиме саморегенерации задержка от момента, когда SDC воспринимает запрос чтения/записи, до момента вызова команды активации банка составляет $2 \times (t_{RAS} + t_{RP})$.

Необходимо отметить, что в режиме саморегенерации генерация сигнала CLKOUT не запрещается контроллером SDRAM. Однако она может быть запрещена программно сбросом бита SCTL5 в регистре EBIU_SDGCTL. В программе необходимо гарантировать, что обращение к адресному пространству SDRAM, вызывающее выход из режима саморегенерации, произойдёт не раньше, чем будут выдержаны все необходимые временные параметры. Если обращение к адресному пространству SDRAM происходит в момент времени, когда бит SCTL5 сброшен, генерируется ошибка внутренней шины и обращение к внешней памяти не производится (таким образом, SDRAM остаётся в режиме саморегенерации). Дополнительную информацию см. в разделе “Обнаружение ошибок”.

17 Интерфейс внешней шины

Команды No Operation/Command Inhibit

Команда “Операция не выполняется” (NOP) не влияет на операции SDRAM, выполняемые в текущий момент времени. Команда “Запрет выполнения команд” (Command Inhibit) аналогична команде NOP; однако при её выполнении выбор кристалла SDRAM не происходит. Когда SDC выполняет доступ к SDRAM и необходимо добавить дополнительные команды, не влияя на текущие операции, даётся команда NOP. Когда SDC не выполняет доступ к SDRAM, даётся команда “Запрет выполнения команд”.

Спецификации временных параметров SDRAM

SDC обеспечивает возможность программирования значений t_{RAS} , t_{RP} , t_{RCD} , t_{WR} и режима выполнения процедуры включения для поддержки ключевых требований к временным параметрам и процедур включения устройств SDRAM различных производителей. Дополнительную информацию см. в разделе “Регистр глобального управления SDRAM (EBIU_SDGCTL)”. Значение задержки CAS в регистре EBIU_SDGCTL должно задаваться в соответствии со значением рабочей частоты (за дополнительной информацией следует обращаться к спецификациям, поставляемым производителями SDRAM).

Остальные параметры, используемые контроллером SDRAM:

- длительность цикла банка: $t_{RC} = t_{RAS} + t_{RP}$;
- длительность цикла регенерации: $t_{RFC} = t_{RAS} + t_{RP}$;
- время выхода из режима саморегенерации: $t_{XSR} = t_{RAS} + t_{RP}$;
- период от команды загрузки регистра режима до команды активации: t_{MRD} или $t_{RSC} = 3$ такта тактового сигнала;
- штраф за промах при обращении к странице = $t_{RP} + t_{RCD}$.

Производительность SDRAM

В таблице 7-2 приведены скорости передачи данных при выполнении ядром или контроллером DMA доступов чтения/записи к 16-разрядной SDRAM. В данном примере предполагается, что все такты равны такту SCLK, и используются следующие значения параметров SDRAM и частоты SCLK:

- частота SCLK = 133 МГц
- задержка CAS = 2 такта ($CL = 2$)
- буферизация SDRAM не выполняется ($EBUFE = 0$)
- Задержка между RAS и предзарядом (t_{RP}) = 2 такта ($TRP = 2$)
- Задержка между RAS и CAS (t_{RCD}) = 2 такта ($TRCD = 2$)
- Время активности команды (t_{RAS}) = 5 тактов ($TRAS = 5$)

Когда используется внешняя буферизация ($EBUFE = 1$ в регистре глобального управления SDRAM) и/или задержка CAS равна 3 ($CL = 11$ в регистре

17 Интерфейс внешней шины

глобального управления SDRAM), при каждом обращении к SDRAM добавляется по одному такту для каждого из перечисленных выше случаев.

Запрос и предоставление доступа к шине

Процессор может передавать управление шинами адреса и данных внешнему устройству. Для того чтобы внешний контроллер мог обращаться к асинхронной или синхронной памяти, процессор переводит свой интерфейс памяти в третье состояние.

Выполнение передачи доступа к шине

Когда внешнее устройство запрашивает доступ к шине, оно выставляет сигнал запроса шины (\overline{BR}). Арбитраж сигнала \overline{BR} выполняется совместно с запросами EAB. Если отсутствуют внутренние запросы, которые необходимо выполнить, внешнему устройству будет предоставлен доступ к внешней шине. Процессор инициирует передачу доступа к шине следующим образом:

- Переводя в третье состояние шины данных, адреса и управляющих сигналов асинхронной памяти. Также могут (необязательно) переводиться в третье состояние управляющие сигналы синхронной памяти.
- Выставляя сигнал предоставления доступа к шине (\overline{BG}).

Процессор может останавливать процесс выполнения программы, если доступ к шине предоставлен внешнему устройству, и формируется запрос чтения/записи или выборки из внешней памяти. Когда внешнее устройство снимает сигнал \overline{BR} , процессор снимает сигнал \overline{BG} и продолжает выполнение программы, с той точки, на которой оно было остановлено.

Когда процессор готов к началу доступа к внешней памяти, но выполнение программы остановлено, так как доступ к шине предоставлен внешнему контроллеру, он выставляет сигнал на выводе \overline{BGH} .

Когда доступ к шине передан внешнему устройству, устанавливается бит BGSTAT в регистре SDSTAT. Процессор может использовать этот бит для проверки состояния шины. Это позволит избежать инициации транзакций, которые могут быть отложены вследствие того, что доступ к шине передан другому устройству.