**9.1 Введение**

Модуль управления включает в себя логику состояния и управления, не адресованную периферийным устройствам или остальным компонентам инфраструктуры устройства. Этот модуль предоставляет интерфейс для управления следующими областями устройства:

- Функциональное мультиплексирование ввода/вывода

- Элементы управления эмуляцией

- Управление устройством и его состояние

- Регистры управления DDR PHY и управления вводом-выводом

- Регистры управления мультиплексированием событий EDMA

**Примечание**. Для записи в регистры модуля управления MPU должен находиться в привилегированном режиме работы и запись не будет работать из пользовательского режима.

**9.2 Функциональное описание**

**9.2.1 Инициализация модуля управления**

Модуль управления реагирует только на внутренний POR и тип устройства. При включении значения после сброса для регистров определяют безопасное состояние устройства. В режиме инициализации , только модули используемые во время загрузки ассоциируются с выводами контроллера. Входы других модулей внутренне связаны, а выходные выводы отключены. После POR программное обеспечение устанавливает регистры функционального мультиплексирования и конфигурации контактных площадок в желаемые значения в соответствии с запрошенной конфигурацией устройства.

Устройства общего назначения (GP) включают недоступные или недоступные функции. Эти недоступные регистры определяют стандартную или фиксированную конфигурацию или поведение устройства.

Битовое поле CONTROL\_STATUS[7:0] SYS\_BOOT отражает состояние контактов sys\_boot, захваченных в POR в модуле PRCM.

**9.2.2 Регистры управления пэдами**

Регистры управления пинами — это 32-битные регистры для управления мультиплексированием сигналов и другими аспектами каждого ввода-вывода. После POR программное обеспечение должно установить регистры функционального мультиплексирования и конфигурации пэда в желаемые значения в соответствии с запрошенной конфигурацией устройства. Конфигурация управляется пинами или группой пинов. Каждый настраиваемый вывод имеет собственный регистр конфигурации для управления pullup/pulldown и для присвоения данному модулю.

В следующей таблице показано общее описание регистра управления пином.

**Таблица 9-1. Описание полей регистра управления пином**

|  |  |  |  |
| --- | --- | --- | --- |
| Bit | Field | Value | Description |
| 31-7 | Reserved |  | Резерв. Чтение возвращает 0. |
| 6 | SLEWCTRL | 0  1 | Выбор более быструю или медленной скорости нарастания.  Быстрый.  Медленный(1) |
| 5 | RXACTIVE | 0  1 | Введите значение разрешения для пина. Установите значение 0 только для вывода. Установите на 1 для ввода или вывода.  Премник деактивирован.  Приемник активирован. |
| 4 | PULLTYPESEL | 0  1 | Выбор типа подтяжки/подтягивания пэда.  Pulldown.  Pullup. |
| 3 | PULLUDEN | 0  1 | Включить Pullup/pulldown у пина.  Pullup/pulldown вкл.  Pullup/pulldown выкл. |
| 2-0 | MUXMODE |  | Выбор мультиплексора функционального сигнала пина |

1. Некоторые периферийные устройства не поддерживают медленную скорость нарастания. Чтобы определить, какие интерфейсы поддерживают каждую скорость нарастания, см. Процессоры AM335x Sitara.(номер публикации SPRS717).

9.2.2.1 Выбор режима

Поле MUXMODE в регистрах управления пэдом определяет режим мультиплексирования, применяемый к пэду. Режимы обозначаются их десятичным (от 0 до 7) или двоичным (от 0b000 до 0b111) представлением. Для большинства пинов значение сброса для поля MUXMODE в регистрах равно 0b111. Исключением являются пины используемые во время загрузки для передачи данных с выбранных периферийных устройств на внешнюю флэш-память.

**Таблица 9-2. Выбор режима**

|  |  |
| --- | --- |
| MUXMODE | Selected Mode |
| 000b | Primary Mode = Mode 0 |
| 001b | Mode 1 |
| 010b | Mode 2 |
| 011b | Mode 3 |
| 100b | Mode 4 |
| 101b | Mode 5 |
| 110b | Mode 6 |
| 111b | Mode 7 |

Режим 0 является основным режимом. Когда установлен режим 0, функция, сопоставленная с выводом, соответствует имени пина. Режимы от 1 до 7 являются возможными режимами для альтернативных функций. На каждом выводе несколько режимов эффективно используются для альтернативных функций, в то время как другие режимы не используются и не соответствуют никаким функциональным конфигурациям.

**Внимание**.

Мультиплексор, управляющий выбором режима сигнала, не является безглючной

структурой. Таким образом, можно увидеть сбой сигнала на несколько наносекунд

во время изменения MUXMODE. Пользователь должен убедиться, что сбой не вызывает

конфликтов или отрицательно влияет на внешнее устройство, подключенное к пину.

**9.2.2.2 Выбор подтяжки**

Отсутствует автоматическое управление стробированием, гарантирующее, что внутренние слабые подтягивающие/подтягивающие резисторы на контактной площадке отключены всякий раз, когда пин сконфигурирован как выход. Если пин всегда сконфигурирован в режиме вывода, он рекомендуется для программного обеспечения пользователя отключать любой внутренний подтягивающий резистор, привязанный к нему, чтобы избежать ненужного потребления. В следующей таблице приведены различные возможные комбинации PULLTYPESEL и поля PULLUDEN управляющего регистра PAD.

**Таблица 9-3. Выбор подтяжки**

|  |  |  |
| --- | --- | --- |
| PULL TYPE | | Pin Behavior |
| PULLTYPESEL | PULLUDENABLE |  |
| 0b | 0b | Pulldown selected and activated |
| 0b | 1b | Pulldown selected but not activated |
| 1b | 0b | Pullup selected and activated |
| 1b | 1b | Pullup selected but not activated |

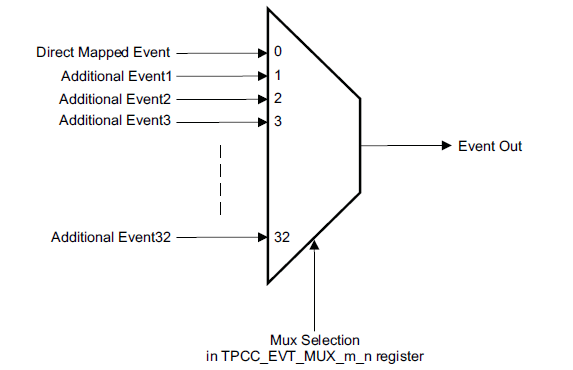
**9.2.2.3 Режим RX Active**

Бит RXACTIVE используется для включения и закрытия входного буфера. Этот элемент управления может быть использован для помощи в утечка питания или изоляция устройства через ввод-вывод. Характеристика сигнала в конечном счете диктуется режимом мультиплексирования, в который помещается пин.

**9.2.3 Мультиплексирование событий EDMA**

Устройство имеет больше событий DMA, чем может разместить максимальное количество TPCC событий, а это 64. Для преодоления в устройстве предусмотрена один мультиплексор на верхнем уровне. Этот модуль будет мультиплексировать дополнительные события со всеми событиями прямого сопоставления. Регистры управления мультиплексором определены в модуле для выбора события маршрутизации на TPCC. Событие с прямым отображением используется по умолчанию (выбор мультиплексора установлен на «0»).

**Мультиплексор событий**



Для каждого события EDMA в конструкции реализовано мультиплексирование, как показано на рисунке.Сопоставленное событие/прерывание всегда будет подключено к входу мультиплексора[0], дополнительные события будут подключены на вход мультиплексора[1], вход мультиплексора[2] и т. д., как определено в таблице событий EDMA. Значение выбора Mux запрограммировано в соответствующий регистр TPCC\_EVT\_MUX\_n. Значение EVT\_MUX может принимать значение от 1 до 32. Остальные значения зарезервированы. По умолчанию значение MUX\_selection равно 0, что означает прямое отображаемое событие подключаемое к выходу Event.

Когда дополнительное событие выбрано с помощью программирования мультиплексора, событие прямого сопоставления не используется.

Например, при событии TINT0 (прерывание таймера 0), которое не связано напрямую с источником события DMA , источник события должен быть подключен к каналу EDMA № 24 (который напрямую привязан к событию SDTXEVT0). Пользователь должен запрограммировать поле EVT\_MUX\_24 в регистре TPCC\_EVT\_MUX\_24\_27 на 22 (значение, соответствующее прерыванию TINT0 в отображении мильтиплексора). Когда это установлено, событие прерывания TINT0 вызовет срабатывание канала 24.

Обратите внимание, что как только это установлено. EDMA больше не может обрабатывать SDTXEVT0. Пользователь должен выделить правильный номер события прямого доступа к памяти для событий с отображением мультиплексора, чтобы не было компрометации на выделение каналов для используемых номеров событий.

**9.2.4 Управление устройством и его состояние**

**9.2.4.1 Состояние управления и загрузки(boot)**

Конфигурация устройства устанавливается во время включения питания или аппаратного сброса (последовательность PORz) конфигурацией входных контактов (SYSBOOT[15:0]). Регистр CONTROL\_STATUS отражает загрузку системы и тип конфигурации устройства, выбранные при подаче сигнала сброса при включении питания (PORz). Входные контакты конфигурации опрашиваются непрерывно в течение активного периода PORz, а в конце выборки значение до последнего нарастающего фронта фиксируется в регистре. Регистр CONTROL\_STATUS дает статус процесса загрузки устройства.

**9.2.4.2 Межпроцессорная коммуникация**

Модуль управления имеет регистры IPC\_MSG\_REG (7:0), которые предназначены для обмена сообщениями между CortexM3 и микропроцессор Cortex A8. Регистр конца события M3 TX (M3\_TXEV\_EOI) обеспечивает механизм чтобы очистить/включить событие TX от Cortex M3 к подсистеме MPU Cortex A8. См. M3\_TXEV\_EOI описание регистра для получения дополнительной информации.

См. Раздел 8.1.4.6, Функциональная последовательность управления питанием с помощью Cortex M3, для получения более подробной информации о том, как регистры IPC\_MSG\_REG используются для связи с прошивкой Cortex-M3.

**9.2.4.3 Управление приоритетом инициатора**

Модуль управления предоставляет регистры для управления приоритетом межсоединения шины и приоритетом EMIF.

**9.2.4.3.1 Управление приоритетом инициатора для межсоединения**

Регистр INIT\_PRIORITY\_n управляет приоритетом инфраструктуры на соединениях шины. Это может быть использовано для динамического повышения приоритета. Существуют битовые поля, которые управляют приоритетом соединения для каждого инициатора шины. По умолчанию всем инициаторам присваивается одинаковый приоритет, и распределение выполняется на основе round robin.

Приоритет может принимать значения от 0 до 3. В следующей таблице приведен допустимый набор значений приоритета.

**Таблица 9-4. Значения приоритета межсоединения**

|  |  |
| --- | --- |
| Interconnect Priority Value | Remarks |
| 00 | **Низкий приоритет** |
| 01 | **Средний приоритет** |
| 10 | **Резерв** |
| 11 | **Высокий приоритет** |

**9.2.4.3.2 Приоритет инициатора в EMIF**

Регистр MREQPRIO предоставляет интерфейс для изменения приоритетов доступа для различных ведущих устройств доступа к EMIF (DDR). Программное обеспечение может использовать этот регистр для установки приоритетов запрашивающей стороны для требуемого арбитража EMIF. Приоритет EMIF может принимать значение от 000b до 111b, где 000b будет самый высокий приоритет, а 111b будет иметь самый низкий приоритет.

**9.2.4.4 Периферийное управление и состояние**

**9.2.4.4.1 Управление и состояние USB**

Регистры USB\_CTRLn и USB\_STSn отражают управление и состояние экземпляров USB. USB линии ввода-вывода могут использоваться как линии UART TX и RX. Битовое поле регистра управления USB GPIOMODE имеет настройки которые настраивают линии USB как линии GPIO. Другие настройки управления USB PHY для управления OTG настройками и PHY являются частью регистра USB\_CTRLn.

Регистр USB\_STSn показывает состояние модуля USB PHY. См. описание регистра USB\_STSn для получения дополнительной информации.

См. Раздел 16.1.4, Детали USB GPIO, для получения дополнительной информации.

**9.2.4.4.2 Обнаружение зарядного устройства USB**

Каждый USB PHY содержит схему, которая может автоматически определять наличие зарядного устройства, подключенного к порту USB. Схема обнаружения зарядного устройства соответствует пересмотренной спецификации зарядки аккумулятора.1.1 с форума разработчиков USB, который можно найти на сайте www.usb.org. Подробнее см. в этом документе подробности о реализации зарядного устройства USB.

**9.2.4.4.2.1 Особенности**

Схема обнаружения зарядного устройства каждого физического уровня имеет следующие особенности:

- Содержит конечный автомат, который может автоматически определять наличие зарядного нисходящего порта или выделенного порт для зарядки (см. определение этих терминов в Спецификации зарядки аккумулятора)

- Выдает сигнал включения зарядного устройства (драйвер CMOS с активным высоким уровнем 3,3 В) при наличии зарядного устройства.

- Позволяет включать/отключать схему для экономии энергии

- Для работы схемы обнаружения требуется только питание 3,3 В.

- Обнаружение зарядного устройства также имеет ручной режим, который позволяет пользователю применить спецификацию зарядки в программном обеспечении.

**9.2.4.4.2.2 Эксплуатация**

Модуль управления предоставляет следующий интерфейс для управления схемой автоматического обнаружения зарядного устройства:

• USB\_CTRLx.CDET\_EXTCTL: включает/выключает автоматическое обнаружение. Оставьте этот бит равным 0, чтобы сохранить включенным автоматическое обнаружение. Изменение этого параметра на 1 включает ручной режим.

• USB\_CTRLx.CHGDET\_RSTRT: перезапускает конечный автомат обнаружения зарядного устройства. Чтобы запустить зарядное устройство обнаружения, измените этот бит с 1 на 0. Если этот бит равен 1, выход включения зарядного устройства (CE) отключен.

• USB\_CTRLx.CHGDET\_DIS: включает/отключает схему обнаружения зарядного устройства. Держите этот бит 0, чтобы сохранить это обнаружение зарядного устройства включено. Установка этого бита в 1 отключит схему обнаружения зарядного устройства.

• USB\_CTRLx.CM\_PWRDN: включает/выключает PHY, который содержит схему обнаружения зарядного устройства. Очистите этот бит в 0, чтобы включить питание PHY.

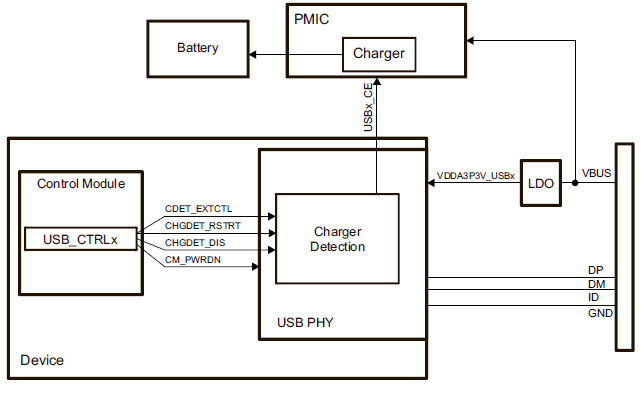
Чтобы начать обнаружение зарядного устройства во время нормальной работы, убедитесь, что PHY и зарядное устройство включены и автоматическое обнаружение включено. Затем инициируйте цикл обнаружения зарядного устройства, перейдя CHGDET\_RSTRT от 1 до 0. Если обнаружен нисходящий порт для зарядки или выделенный порт для зарядки, сигнал включения зарядного устройства (USBx\_CE) будет иметь высокий уровень и оставаться высоким до тех пор, пока зарядное устройство не будет отключено

либо CHGDET\_DIS = 1, либо CHGDET\_RSTRT = 1. Если порт остается неподключенным после запуска цикла обнаружения зарядного устройства, он будет продолжать обнаружение до тех пор, пока не будет обнаружено зарядное устройство или не возникнет состояние ошибки.

Обратите внимание, что USBx\_CE не является выходом с открытым стоком.

Чтобы отключить зарядное устройство после успешного обнаружения, вы должны отключить схему обнаружения зарядного устройства с помощью CHGDET\_DIS или CHGDET\_RSTRT, даже если зарядное устройство физически отключено.

**Рисунок 9-1. Обнаружение зарядного устройства USB**



Обнаружение зарядного устройства может быть запущено автоматически при отсутствии питания остальной части AM335x. Если VDDA3P3V\_USBx присутствует, через LDO с питанием от VBUS, подключенного к хосту, конечный автомат обнаружения зарядного устройства будет автоматически запускать и выполнять обнаружение. Если зарядное устройство обнаружено, USBx\_CE будет переведен в высокий уровень, в противном случае это будет переведен в низкий.

Схема обнаружения зарядного устройства выполняет следующие шаги спецификации зарядки аккумулятора версии 1.1:

1. Обнаружение VBUS

2. Обнаружение контакта с данными

3. Первичное обнаружение

Вторичное обнаружение (чтобы различать зарядный нисходящий порт и выделенный зарядный порт Port) — это недавно добавленная функция спецификации v1.2, которая не реализована в конечном автомате обнаружения зарядки.

**ПРИМЕЧАНИЕ**. Выход USBx\_CE будет работать только тогда, когда соответствующий контакт USBx\_ID заземлен (указывает на режим хоста USB). Выход USBx\_CE не работает в периферийном режиме (когда USBx\_ID плавающий).

**9.2.4.4.4 Управление изоляцией сброса модуля Ethernet**

Эта функция позволяет устройству выполнить «теплый» сброс без нарушения работы коммутатора или маршрутизации трафика через переключатель во время состояния сброса. Регистр изоляции сброса CPSW (RESET\_ISO) имеет поле ISO\_CONTROL, которое управляет функцией изоляции сброса.

Если изоляция сброса включена, любой источник теплого сброса будет заблокирован для коммутатора EMAC. Если EMAC изоляция сброса НЕ активна (состояние по умолчанию), тогда источникам теплого сброса разрешено распространяться как нормальному, в том числе к модулю коммутатора EMAC (оба входа сброса на IP). Все холодные сбросы или сбросы POR будут всегда распространяется на модуль коммутатора EMAC как обычно.

Когда RESET\_ISO включен, следующие регистры не будут затронуты теплым сбросом:

- GMII\_SEL

- CONF\_GPMC\_A[11:0]

- CONF\_GPMC\_WAIT0

- CONF\_GPMC\_WPN

- CONF\_GPMC\_BEN1

- CONF\_MII1\_COL

- CONF\_MII1\_CRS

- CONF\_MII1\_RX\_ER

- CONF\_MII1\_TX\_EN

- CONF\_MII1\_RX\_DV

- CONF\_MII1\_TXD[3:0]

- CONF\_MII1\_TX\_CLK

- CONF\_MII1\_RX\_CLK

- CONF\_MII1\_RXD[3:0]

- CONF\_RMII1\_REF\_CLK

- CONF\_MDIO

- CONF\_MDC

**9.2.4.4.5 Управление захватом событий таймера/eCAP**

События Timer 5, 6, 7 и события eCAP0, 1, 2 можно выбрать с помощью TIMER\_EVT\_CAPTURE. и регистров ECAP\_EVT\_CAPTURE. В следующей таблице перечислены доступные источники этих событий.

**Таблица 9-5. Доступные источники для событий Timer[5–7] и eCAP[0–2]**

|  |  |  |
| --- | --- | --- |
| Event No. | Source module | Interrupt Name/Pin |
| 0 | For Timer 5 MUX input from IO signal  TIMER5 | TIMER5 IO pin |
| For Timer 6 MUX input from IO signal  TIMER6 | TIMER6 IO pin |
| For Timer 7 MUX input from IO signal  TIMER7 | TIMER7 IO pin |
| For eCAP 0 MUX input from IO signal  eCAP0 | eCAP0 IO pin |
| For eCAP 1 MUX input from IO signal  eCAP1 | eCAP1 IO pin |
| For eCAP 2 MUX input from IO signal  eCAP2 | eCAP2 IO pin |
| 1 | UART0 | UART0INT |
| 2 | UART1 | UART1INT |
| 3 | UART2 | UART2INT |
| 4 | UART3 | UART3INT |
| 5 | UART4 | UART4INT |
| 6 | UART5 | UART5INT |
| 7 | 3PGSW | 3PGSWRXTHR0 |
| 8 | 3PGSW | 3PGSWRXINT0 |
| 9 | 3PGSW | 3PGSWTXINT0 |
| 10 | 3PGSW | 3PGSWMISC0 |
| 11 | McASP0 | MCATXINT0 |
| 12 | McASP0 | MCARXINT0 |
| 13 | McASP1 | MCATXINT1 |
| 14 | McASP1 | MCARXINT1 |
| 15 | Reserved | Reserved |
| 16 | Reserved | Reserved |
| 17 | GPIO 0 | GPIOINT0A |
| 18 | GPIO 0 | GPIOINT0B |
| 19 | GPIO 1 | GPIOINT1A |
| 20 | GPIO 1 | GPIOINT1B |
| 21 | GPIO 2 | GPIOINT2A |
| 22 | GPIO 2 | GPIOINT2B |
| 23 | GPIO 3 | GPIOINT3A |
| 24 | GPIO 3 | GPIOINT3B |
| 25 | DCAN0 | DCAN0\_INT0 |
| 26 | DCAN0 | DCAN0\_INT1 |
| 27 | DCAN0 | DCAN0\_PARITY |
| 28 | DCAN1 | DCAN1\_INT0 |
| 29 | DCAN1 | DCAN1\_INT1 |
| 30 | DCAN1 | DCAN1\_PARITY |

**9.2.4.4.6 Управление захватом АЦП**

Следующие события уровня микросхемы могут быть подключены через программно-управляемый мультиплексор к модулю TSC\_ADC.

1. Хост-событие PRU-ICSS 0

2. Таймер 4 Событие

3. Таймер 5 Событие

4. Таймер 6 Событие

5. Таймер 7 Событие

Этот контакт является внешним аппаратным триггером для запуска преобразования канала АЦП. Регистр ADC\_EVT\_CAPT необходимо запрограммировать, чтобы выбрать правильный источник для этого преобразования.

**События Таймера**

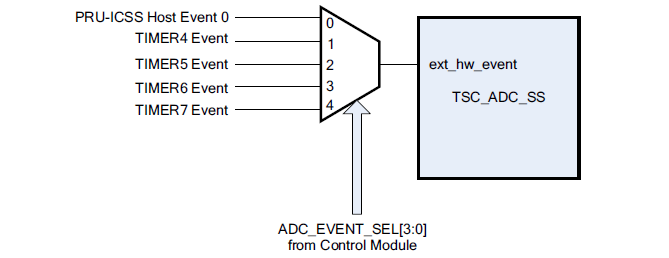
****

Таблица 9-6 содержит значение, которое необходимо запрограммировать в мультиплексоре выбора.

**Таблица 9-6. Выбор значений мультиплексора**

|  |  |
| --- | --- |
| ADC\_EVENT\_SEL Value | ADC External event selected |
| 0b000 | PRU-ICSS Host Event 0 |
| 0b001 | Timer 4 Event |
| 0b010 | Timer 5 Event |
| 0b011 | Timer 6 Event |
| 0b100 | Timer 7 Event |
| 0b101-111 | Reserved |

**9.2.4.4.7 Управление LDO SRAM**

Устройство включает в себя два экземпляра модуля SRAM LDO (VSLDO). Один из этих LDO питает внутреннюю SRAM ARM, а другая питает SRAM OCMC. В регистре SMA2 бит VSLDO\_CORE\_AUTO\_RAMP\_EN, если он установлен, позволяет VSLDO, который питает SRAM OCMC, сохраняться во время глубокого сна и обеспечивать более низкое энергопотребление. Поскольку VSLDO является общим между памятью WKUP M3 и памятью CORE, VSLDO должен быть выведен из хранения на любое событие пробуждения. Этот бит разрешает эту функциональность и должен быть установлен для обеспечения правильного сна/пробуждения при работе в режимах Standby и DeepSleep. Подобная функциональность не нужна для питания внутренней SRAM ARM от LDO. Его можно перевести в режим хранения с помощью PRM\_LDO\_SRAM\_MPU\_CTRL.

**9.2.5 Физическая память DDR**

**Таблица 9-7. Настройки управления скоростью нарастания DDR**(1) (2)

|  |  |  |
| --- | --- | --- |
| sr1 | sr0 | Slew Rate Level |
| 0 | **0** | Fastest |
| 1 | **0** | Fast |
| 0 | **1** | Slow |
| 1 | **1** | Slowest |

(1) Эти значения запрограммированы в следующих регистрах: ddr\_cmd0\_ioctrl, ddr\_cmd1\_ioctrl, ddr\_cmd2\_ioctrl, ddr\_data0\_ioctrl, ddr\_data1\_ioctrl.

(2) Значения для DDR\_CMDx\_IOCTRL.io\_config\_sr\_clk должны быть запрограммированы на одно и то же значение.

**Таблица 9-8. Настройки управления импедансом DDR**(1) (2) (3)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| I2 | I1 | I0 | Output Impedance  (Ron) | Drive Strength  |IOH|, |IOL| | Example:  Ron for Rext =  49.9 ohms | Example:  |IOH|, |IOL| for Rext =  49.9 ohms |
| 0 | **0** | **0** | 1.6\*Rext | 0.625\*Iout | 80 ohms | 5 mA |
| 0 | **0** | **1** | 1.33\*Rext | 0.75\*Iout | 67 ohms | 6 mA |
| 0 | **1** | **0** | 1.14\*Rext | 0.875\*Iout | 57 ohms | 7 mA |
| 0 | **1** | **1** | Rext | Iout | 50 ohms | 8 mA |
| 1 | **0** | **0** | 0.88\*Rext | 1.125\*Iout | 44 ohms | 9 mA |
| 1 | **0** | **1** | 0.8\*Rext | 1.250\*Iout | 40 ohms | 10 mA |
| 1 | **1** | **0** | 0.73\*Rext | 1.375\*Iout | 36 ohms | 11 mA |
| 1 | **1** | **1** | 0.67\*Rext | 1.5\*Iout | 33 ohms | 12 mA |

(1) Эти значения запрограммированы в следующих регистрах: ddr\_cmd0\_ioctrl, ddr\_cmd1\_ioctrl, ddr\_cmd2\_ioctrl, ddr\_data0\_ioctrl,ddr\_data1\_ioctrl.

(2) Значения для DDR\_CMDx\_IOCTRL.io\_config\_i\_clk должны быть запрограммированы на одно и то же значение.

(3) Rext — это внешний компенсационный резистор VTP, подключенный к клемме DDR\_VTP.

**9.2.5.1 Сопоставление контактов DDR PHY и IO**

В следующей таблице описывается сопоставление контактов DDR PHY и IO.

**Таблица 9-9. Сопоставление контактов DDR PHY и IO**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Macro Pin | CMD0 | CMD1 | CMD2 | DATA0 | DATA1 |
| 0 | ddr\_ba2 | Unconn | ddr\_cke | ddr\_d8 | ddr\_d0 |
| 1 | ddr\_wen | ddr\_a15 | ddr\_resetn | ddr\_d9 | ddr\_d1 |
| 2 | ddr\_ba0 | ddr\_a2 | ddr\_odt | ddr\_d10 | ddr\_d2 |
| 3 | ddr\_a5 | ddr\_a12 | Unconn | ddr\_d11 | ddr\_d3 |
| 4 | ddr\_ck | ddr\_a7 | ddr\_a14 | ddr\_d12 | ddr\_d4 |
| 5 | ddr\_ckn | ddr\_ba1 | ddr\_a13 | ddr\_d13 | ddr\_d5 |
| 6 | ddr\_a3 | ddr\_a10 | ddr\_csn0 | ddr\_d14 | ddr\_d6 |
| 7 | ddr\_a4 | ddr\_a0 | Unconn | ddr\_d15 | ddr\_d7 |
| 8 | ddr\_a8 | ddr\_a11 | ddr\_a1 | ddr\_dqm1 | ddr\_dqm0 |
| 9 | ddr\_a9 | ddr\_casn | Unconn | ddr\_dqs1 | ddr\_dqs0 |
| 10 | ddr\_a6 | ddr\_rasn | Unconn | ddr\_dqsn1 | ddr\_dqsn0 |