**19.Universal Asynchronous Receiver/Transmitter (UART)**

**19.1 Введение**

**19.1.1 Фичи режима UART**

Общие характеристики модуля UART/IrDA при работе в режиме UART:

- Совместимость с 16C750.

- Скорость передачи от 300 бит/с до 3,6864 Мбит/с.

- Автопереключение между 1200 бит/с и 115,2 кбит/с.

- Программное/аппаратное управление потоком.

- Программируемые символы Xon/Xoff.

– Программируемый Auto-RTS и Auto CTS.

- Характеристики программируемого последовательного интерфейса.

– 5, 6, 7 или 8-битные символы.

– Генерация четных, нечетных, меток (всегда 1), пробелов (всегда 0) или битов без четности (битовый кадр без четности) и обнаружения.

– Генерация 1, 1,5 или 2 стоповых бита.

- Обнаружение ложного стартового бита.

- Генерация и обнаружение разрыва строки.

- Функции управления модемом (CTS, RTS, DSR, DTR, RI и DCD).

- Управление системой прерываний с полным приоритетом.

- Возможности внутреннего тестирования и замыкания на себя.

**19.1.2 Особенности режима ИК-порта(IrDA)**

Общие характеристики **UART/IrDA** при работе в режиме **IrDA**:

- Поддержка связи **IrDA** 1.4 в медленном инфракрасном диапазоне (**SIR**), среднем инфракрасном диапазоне (**MIR**) и быстром инфракрасном диапазоне (**FIR**).

(очень быстрый инфракрасный порт (**VFIR**) не поддерживается).

- Форматирование кадра: добавление переменных символов **xBOF** и символов **EOF**.

- Генерация/обнаружение **CRC** восходящей/нисходящей линии связи.

- Асинхронная прозрачность (автоматическая вставка символа разрыва).

- **FIFO** состояния на 8 записей (с выбираемыми уровнями срабатывания), доступный для контроля длины кадра и ошибок кадра.

- Ошибка кадрирования, ошибка проверки циклическим избыточным кодом (**CRC**), недопустимый символ (**FIR**), шаблон прерывания (**SIR**, **MIR**).

**19.1.3 Особенности режима CIR**

Общие характеристики **UART**/**IrDA** при работе в режиме **CIR**:

- Поддержка потребительского инфракрасного излучения (**CIR**) для приложений дистанционного управления.

- Передача и получение.

- Свободный формат данных (поддерживает любые частные стандарты удаленного управления).

- Выбираемая скорость передачи данных.

- Настраиваемая несущая частота.

- Рабочий цикл 1/2, 5/12, 1/3 или 1/4 несущей.

**19.1.4 Неподдерживаемые функции UART**

Следующие функции модуля **UART**/**IrDA** не поддерживаются этим устройством:

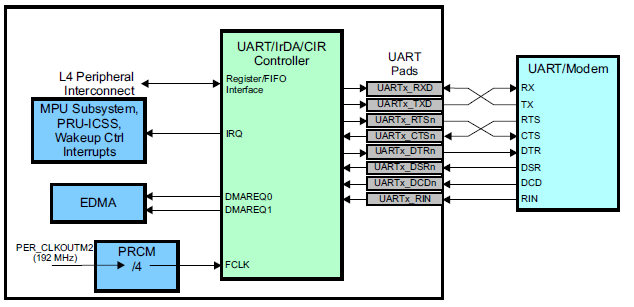
**Таблица 19-1. Неподдерживаемые функции UART**

|  |  |
| --- | --- |
| Feature | Reason |
| Full modem control on UART0 | DCD, DSR, DTR, RI not pinned-out |
| Full modem control on UART2-5 | DCD, DSR, DTR, RI not pinned-out |
| Device wake-up on UART1-5 | Wake-up not supported - no SWake connection |
| DMA mode 2 and 3 | Only DMA mode 0 and 1 is supported with EDMA |

**19.2 Интеграция**

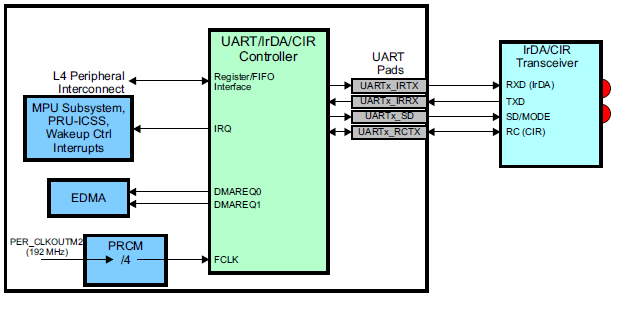
Это устройство содержит 6 экземпляров периферийного устройства **UART**/**IrDA** (**UARTIRDAOCP**). Есть шесть **UART** модулей, они называются **UART0** – **UART5**. **UART0** обеспечивает возможность пробуждения. Только **UART1** обеспечивает управляющие сигналы полноценного модема. Все **UART** поддерживают режимы **IrDA** и **CIR**, а также управление потоком **RTS**/**CTS** (с учетом мультиплексирования контактов конфигурации). На **рис. 19-1** показан пример соединения системы с использованием связи **UART** с аппаратным рукопожатием.

**Рисунок 19-1. Модуль UART/IrDA — Приложение UART**



На рис. 19-2 показан пример подключения системы с помощью инфракрасной связи с дистанционным управлением.(бытовой инфракрасный).

**Рисунок 19-2. Модуль UART/IrDA — Приложение IrDA/CIR**



**19.2.1 Атрибуты подключения UART**

Общие атрибуты подключения для каждого из модулей UART показаны в Таблице 19-2 и Таблице 19-3.

**Таблица 19-2. Атрибуты подключения UART0**

|  |  |
| --- | --- |
| Attributes | Type |
| Power Domain | Wake-Up Domain |
| Clock Domain | PD\_WKUP\_L4\_WKUP\_GCLK (OCP)  PD\_WKUP\_UART0\_GFCLK (Func) |
| Reset Signals | WKUP\_DOM\_RST\_N |
| Idle/Wakeup Signals | Smart Idle / Wakeup |
| Interrupt Requests | 1 interrupt to MPU Subsystem (UART0INT), PRU-ICSS (nirq)  and WakeM3 |
| DMA Requests | 2 DMA requests to EDMA (TX – UTXEVT0, RX – URXEVT0) |
| Physical Address | L4 Wakeup slave port |

**Таблица 19-3. Атрибуты подключения** **UART1-5**

|  |  |
| --- | --- |
| Attributes | Type |
| Power Domain | Peripheral Domain |
| Clock Domain | PD\_PER\_L4LS\_GCLK (OCP)  PD\_PER\_UART\_GFCLK (Func) |
| Reset Signals | PER\_DOM\_RST\_N |
| Idle/Wakeup Signals | Smart Idle |
| Interrupt Requests | UART1-2  1 interrupt per instance to MPU Subsystem (UART1INT,  UART2INT) and PRU-ICSS (nirq)  UART3-5  1 interrupt per instance to only MPU Subsystem (UART3INT,  UART4INT, UART5INT) |
| DMA Requests | 2 DMA requests per instance to EDMA (TX – UTXEVTx, RX –  URXEVTx) |
| Physical Address | L4 Peripheral slave port |

**19.2.2 Управление синхронизацией и сбросом UART**

Модули UART используют отдельные функциональные и шинные интерфейсы тактирования.

**Таблица 19-4. Сигналы тактирования UART0**

|  |  |  |  |
| --- | --- | --- | --- |
| Clock Signal | Max Freq | Reference / Source | Comments |
| CLK  Interface clock  From PRCM | 100 MHz | CORE\_CLKOUTM4 / 2 | pd\_wkup\_l4\_wkup\_gclk |
| FCLK  Functional clock  From PRCM | 48 MHz | PER\_CLKOUTM2 / 4 | pd\_wkup\_uart0\_gfclk |

**Таблица 19-5. UART1-5 Тактовые сигналы**

|  |  |  |  |
| --- | --- | --- | --- |
| Clock Signal | Max Freq | Reference / Source | Comments |
| CLK  Interface clock | 100 MHz | CORE\_CLKOUTM4 / 2 | pd\_per\_l4ls\_gclk  From PRCM |
| FCLK  Functional clock | 48 MHz | PER\_CLKOUTM2 / 4 | pd\_per\_uart\_gfclk  From PRCM |

Для работы UART функциональное тактирование используются для обеспечения скорости передачи данных до 3,6 Мбит/с. Табл. 19-6 показан список настройки скоростей передачи от делителя частоты

**Таблица 19-6. Baud-rate UART и частоты ошибок**

|  |  |  |  |
| --- | --- | --- | --- |
| Baud rate | Over sampling | Divisor | Error (%) |
| 300 | **16** | **10000** | **0** |
| 600 | **16** | **5000** | **0** |
| 1200 | **16** | **2500** | **0** |
| 2400 | **16** | **1250** | **0** |
| 4800 | **16** | **625** | **0** |
| 9600 | **16** | **313** | **0.16** |
| 14400 | **16** | **208** | **0.16** |
| 19200 | **16** | **156** | **0.16** |
| 28800 | **16** | **104** | **0.16** |
| 38400 | **16** | **78** | **0.16** |
| 57600 | **16** | **52** | **0.16** |
| 115200 | **16** | **26** | **0.16** |
| 230400 | **16** | **13** | **0.16** |
| 460800 | **13** | **8** | **0.16** |
| 921600 | **13** | **4** | **0.16** |
| 1843200 | **13** | **2** | **0.16** |
| 3000000 | **16** | **1** | **0** |
| 3686400 | **13** | **1** | **0.16** |

Для работы IrDA внутренний функциональный делитель тактовых сигналов позволяет генерировать скорости SIR, MIR или FIR в бодах как показано в таблице 19-7.

**Таблица 19-7. Скорость передачи в режиме IrDA и частота ошибок**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Baud rate | IR mode | Encoding | Divisor | Error (%) |
| 2400 | SIR | 3/16 | 1250 | 0 |
| 9600 | SIR | 3/16 | 312 | 0.16 |
| 19200 | SIR | 3/16 | 156 | 0.16 |
| 38400 | SIR | 3/16 | 78 | 0.16 |
| 57600 | SIR | 3/16 | 52 | 0.16 |
| 115200 | SIR | 3/16 | 26 | 0.16 |
| 576000 | MIR | 1/4 | 2 | 0 |
| 1152000 | MIR | 1/4 | 1 | 0 |
| 4000000 | FIR | 4PPM | 1 | 0 |

**19.2.3 Список контактов UART**

Контакты интерфейса UART перечислены в таблице 19-8. Функциональность контактов зависит от выбранного режима модуля.

**Таблица 19-8. Список выводов UART**

|  |  |  |
| --- | --- | --- |
| Pin | Type | Description |
| UARTx\_RXD / IRRX / RCRX | I | UART / IrDA / CIR Receive Data |
| UARTx\_TXD / IRTX / RCTX | OZ | UART / IrDA / CIR Transmit Data |
| UARTx\_RTSn / SD | OZ | UART Request to Send / IrDA Mode |
| UARTx\_CTSn | I | UART Clear to Send |
| UARTx\_DTRn(1) | OZ | UART Data Terminal Ready |
| UARTx\_DSRn(1) | I | UART Data Set Ready |
| UARTx\_DCDn(1) | I | UART Data Carrier Detect |
| UARTx\_RIn(1) | I | UART Ring Indicator |

(1) UART1 only

Модуль UART может работать в трех различных режимах на основе MODE\_SELECT битов. Сигнал мультиплексирование на основе этих битов режима показано в таблице 19-9.

**Таблица 19-9. Управление мультиплексированием UART**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| UARTx\_TXD / IRTX /  RCTX Function | UARTx\_RXD / IRRX / RCRX  Function | UARTx\_RTSn / SD  Function | UARTx\_CTSn  Function | Mode |
| TXD | RXD | RTSn | CTSn | UART |
| IRTX | IRRX | SD | not used | IrDA (SIR, MIR, FIR) |
| RCTX | RCRX | SD | not used | CIR |

**19.3 Функциональное описание**

**19.3.1 Блок-схема**

Модуль **UART/IrDA/CIR** можно разделить на три основных блока:

- Управление FIFO

- Выбор режима

- Форматирование протокола

Управление FIFO является общим для всех функций и позволяет передавать и получать данные с позиции хост-процессора.

Есть два режима:

- Функциональный режим: направляет данные в выбранную функцию (**UART**, **IrDA** или **CIR**) и включает

механизм, соответствующий выбранной функции

- Режим регистрации: разрешает условный доступ к регистрам.

Для получения дополнительной информации о настройке режима см. Раздел 19.3.7, Выбор режима.

Форматирование протокола имеет три подкатегории:

- Генерация тактового сигнала: Входной тактовый сигнал 48 МГц генерирует все необходимые тактовые сигналы.

- Форматирование данных: каждая функция использует свой конечный автомат, отвечающий за переход

между данными FIFO и данными кадра, связанными с ним.

- Управление прерываниями: в зависимости от выбранной функции генерируются различные типы прерываний:

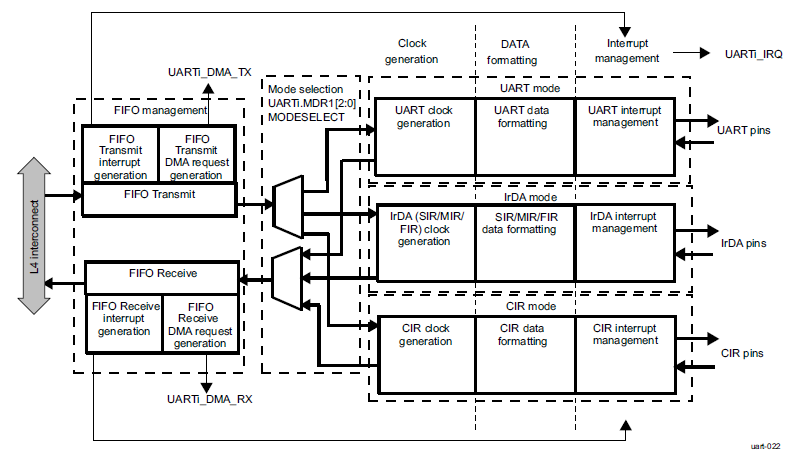
- Прерывания режима **UART**: семь прерываний с приоритетом на шести различных уровнях.

- Прерывания в режиме **IrDA**: восемь прерываний. Линия прерывания активируется при генерации любого прерывания (нет приоритета).

– Прерывания режима **CIR**: используется подмножество существующих прерываний режима **IrDA**.

В каждом режиме при генерации прерывания регистр **UART\_IIR** указывает тип прерывания. Параллельно с этими функциональными блоками для каждой функции существует стратегия энергосбережения. На рис. 19-3 представлена ​​блок-схема **UART/IrDA/CIR**.

**Рис. 19-3. Блок-схема функциональных спецификаций UART/IrDA/CIR**



**19.3.2 Конфигурация тактирования**

Каждый **UART** использует тактовую частоту 48 МГц для своей логики и для генерации сигналов внешнего интерфейса. Каждый **UART** использует интерфейсное тактирование для доступа к регистрам. Модуль **PRCM** генерирует и контролирует все эти тактирования (дополнительную информацию см. в разделе Атрибуты модуля **Clock Domain** в Главе 8, Питание, Сброс и управление тактированием). Процессы бездействия и пробуждения используют протокол квитирования между **PRCM** и **UART** (для описание протокола см. в разделе «Управление тактированием на уровне модуля» в главе 8 «Питание, сброс и управление тактированием"). Битовое поле **UARTi.UART\_SYSC[4:3] IDLEMODE** управляет режимом ожидания **UART**.

**19.3.3 Программный сброс**

Бит **UARTi.UART\_SYSC[1] SOFTRESET** управляет программным сбросом; установка этого бита в **1** запускает программный сброс функционально эквивалентеный аппаратному сбросу.

**19.3.4 Управление питанием**

**19.3.4.1 Управление питанием в режиме UART**

**19.3.4.1.1 Энергосбережение модуля**

В режимах **UART** спящий режим включается установкой бита **UARTi.UART\_IER[4] SLEEP\_MODE** в **1** (когдабит **UARTi.UART\_EFR[4] ENHANCED\_EN** установлен в 1).

Переход в спящий режим осуществляется при соблюдении всех следующих условий:

- Линия последовательного ввода данных, **uarti\_rx**, не занята.

- **TX** **FIFO** и сдвиговый регистр **TX** пусты.

- **RX** **FIFO** пуст.

- Единственными ожидающими прерываниями являются прерывания **THR**.

Спящий режим — хороший способ снизить энергопотребление **UART**, но это состояние может быть достигнуто только если **UART** установлен в режим модема. Поэтому, даже если **UART** не задействован, он должен инициализироваться в функциональном режиме, чтобы использовать спящий режим. В спящем режиме тактирование модуля и тактирование битрейта передачи останавливаются . Поскольку большинство регистров тактируется этими источниками, это значительно снижает энергопотребление. Модуль просыпается при изменении  на линии **uarti\_rx**, при записи данных в **TX FIFO** и при изменении состояние входных контактов модема. Прерывание может быть сгенерировано по событию пробуждения, установив **UARTi.UART\_SCR[4] RX\_CTS\_WU\_EN** бит в **1**. Чтобы понять, как управлять прерыванием, см. Раздел 19.3.5.2, Прерывание пробуждения.

***ПРИМЕЧАНИЕ. Не следует записывать что либо в защелки делителя, UARTi.UART\_DLL и UARTi.UART\_DLH, для установки тактирования битрейта (BCLK) в спящем режиме. Желательно сначала отключить спящий режим с помощью бита UARTi.UART\_IER[4] SLEEP\_MODE перед записью в регистр UARTi.UART\_DLL или регистр UARTi.UART\_DLH.***

**19.3.4.1.2 Энергосбережение системы**

Спящий режим и автоматический режим ожидания — это встроенные функции энергосбережения. Методы снижения мощности могут применяется на системном уровне путем отключения определенного внутреннего тактирования и доменов питания устройства. **UART** поддерживает протокол квитирования запроса/подтверждения бездействия, используемый на системном уровне для отключения **UART** синхронизируется чистым и контролируемым образом, а также переключает **UART** из режима генерации прерываний в режим генерации пробуждения для немаскированных событий (см. бит **UARTi.UART\_SYSC[2] ENAWAKEUP** и регистр **UARTi.UART\_WER**).

Дополнительные сведения см. в разделе «Управление чтактированием на уровне модуля» в главе 8 «Питание, сброс и управление тактированием".

**19.3.4.2 Управление питанием в режиме IrDA/CIR**

**19.3.4.2.1 Энергосбережение модуля**

В режимах **IrDA**/**CIR** спящий режим включается установкой бита **UARTi.MDR[3] IR\_SLEEP** в **1**.Переход в спящий режим осуществляется при соблюдении всех следующих условий:

- Линия ввода последовательных данных, uarti.rx\_irrx, не занята.

- TX FIFO и сдвиговый регистр TX пусты.

- RX FIFO пуст.

- Нет ожидающих прерываний, кроме прерываний THR.

Модуль просыпается при обнаружении изменения в строке **uarti\_rx\_irrx** или при записи данных в **TX FIFO**.

**19.3.4.2.2 Энергосбережение системы**

Энергосбережение системы для режима **IrDA**/**CIR** имеет ту же функцию, что и для режима **UART** (см.Раздел 19.3.4.1.2, Энергосбережение системы).

**19.3.4.3 Локальное управление питанием**

В Таблице 19-10 описаны функции управления питанием, доступные для **UART**.

***ПРИМЕЧАНИЕ. Информацию о стробировании исходного тактового сигнала и описании переходов в спящий/пробуждающий режим см.***

***Управление тактированием на уровне модуля в Главе 8, Управление питанием, сбросом и тактированием.***

**Таблица 19-10. Локальные функции управления питанием**

|  |  |  |
| --- | --- | --- |
| Feature | Registers | Description |
| Clock autogating | UART\_SYSC[0] AUTOIDLE | Этот бит обеспечивает локальную оптимизацию мощности в модуле посредством стробирования UARTi\_ICLK интерфейсного тактирования или стробирования UARTi\_FCLK функционального тактирования внутренней активности. |
| Slave idle modes | UART\_SYSC[4:3] IDLEMODE | Доступны: режим форсированного простоя, режим отсутствия простоя, режим умного простоя и режим пробуждения с возможностью умного простоя. |
| Clock activity | N/A | Функция недоступна |
| Master standby  modes | N/A | Функция недоступна |
| Global wake-up  enable | UART\_SYSC[2] ENAWAKEUP | Этот бит активизирует функцию пробуждения на уровне модуля. |
| Wake-Up sources  enable | N/A | Функция недоступна |

**19.3.5 Запросы на прерывание**

Модуль **UART** **IrDA** **CIR** генерирует прерывания. Все прерывания могут быть разрешены/запрещены путем записи соотвествтующего бита в регистр разрешения прерываний (**IER**). Статус прерывания устройства можно проверить в любое время, читая регистр идентификации прерывания (**IIR**). Режимы **UART**, **IrDA** и **CIR** имеют

разные прерывания в модуле **UART** **IrDA** **CIR** и, следовательно, имеют разные отображения **IER** и **IIR** в соответствии с выбранным режимом.

**19.3.5.1 Управление прерываниями режима UART**

**19.3.5.1.1 Прерывания UART**

Режим **UART** включает семь возможных прерываний с шестью уровнями приоритета. Когда генерируется прерывание, регистр идентификации прерывания (**UARTi.UART\_IIR**) устанавливает **UARTi.UART\_IIR[0]** бит **IT\_PENDING** равен **0**, чтобы указать, что прерывание ожидает обработки, и указывает тип прерывания.

прерывание через битовое поле **UARTi.UART\_IIR[5:1]**. Таблица 19-11 суммирует управление фунциями перываний.

**Таблица 19-11. Прерывания режима UART**