

13 КОНТРОЛЛЕР ПОРТА UART

Универсальный асинхронный приёмопередатчик (UART, Universal Asynchronous Receiver Transmitter) – это полнодуплексное периферийное устройство, совместимое с интерфейсами UART промышленного стандарта. UART преобразует данные между последовательным и параллельным форматами. Последовательная передача (приём) выполняется по асинхронному протоколу, поддерживающему различные длины слов данных, различное количество стоп-битов и возможность формирования битов проверки чётности. Порт UART также включает аппаратные средства обработки прерываний. Прерывания могут генерироваться для 12 различных событий.

Порт UART поддерживает полудуплексный протокол SIR IRDA[®] (Infrared Data Association) со скоростями передач 9.6/115.2 Кбит/с.



Аппаратная поддержка функций управления и сигналов состояния модема в модуле UART отсутствует, однако она может быть реализована с помощью выводов I/O общего назначения (GPIO).

Порт UART является DMA-совместимым устройством, поддерживающим отдельные каналы DMA приёма (RX) и передачи (TX) в режиме ведущего. Он может использоваться либо в режиме DMA, либо в режиме программируемого ввода/вывода (без DMA). В режиме программируемого ввода/вывода требуется программное управление процессом передач данных с помощью прерываний или опроса флагов. В режиме DMA требуется минимальное вмешательство программы, так как контроллер DMA самостоятельно перемещает данные. Дополнительную информацию о DMA см. в главе 9, “Прямой доступ к памяти”.

Любой из периферийных таймеров может использоваться для обеспечения аппаратно поддерживаемой функции определения скорости двоичной передачи UART. Дополнительную информацию см. в главе 15, “Таймеры”.

Последовательная связь

UART работает в соответствии с протоколом асинхронной последовательной связи, имеющим следующие характеристики:

- количество битов данных от 5 до 8;
- 1, 1½ или 2 стоп-бита;
- проверка на чётность, нечётность и без проверки;
- скорость передачи = $SCLK / (16 \times \text{делитель})$, где SCLK – частота тактового сигнала системы, и делитель может принимать значения от 1 до 65536.

Для всех слов данных требуется старт-бит и, по меньшей мере, один стоп-бит. В результате, с учётом необязательного бита чётности, длина слова данных может принимать значения от 7 до 12 битов. Формат принимаемых и передаваемых кадров определяется регистром управления линией связи (UART_LCR). Данные всегда передаются и принимаются, начиная с младшего бита.

На рис. 13-1 показан типичная последовательность битов, наблюдаемая на выводе TX:

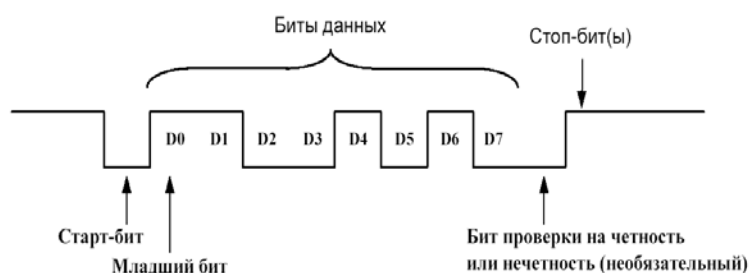


Рис. 13-1. Последовательность битов на выводе TX

Регистры управления и состояния UART

В процессоре имеется набор регистров управления и состояния для каждого порта UART, аналогичный набору регистров интерфейса UART промышленного стандарта. Каждый из этих регистров, отображённых в карте памяти, является восьмиразрядным и занимает половину слова; старший байт слова заполнен нулями.

Аналогично UART-интерфейсам промышленного стандарта несколько регистров порта отображаются в одной ячейке памяти. Адреса регистров делителя (UART_DHL и UART_DLL) совпадают с адресами регистра буфера передачи (UART_THR), регистра буфера приёма (UART_RBR) и регистра разрешения прерываний (UART_IER). Бит доступа к регистру делителя (DLAB) в регистре управления линией связи (UART_LCR) определяет набор регистров, доступный в данный момент времени. Для обращения к этим регистрам в программе должны использоваться команды загрузки/сохранения 16-разрядных слов.

И в приёмном и в передающем канале данные буферизуются. Регистр UART_THR является буферным для сдвигового регистра передачи (TSR), регистр UART_RBR является буферным для сдвигового регистра приёма (LSR). Сдвиговые регистры недоступны программе напрямую.

Регистр UART_LCR

Регистр управления линией связи UART (UART_LCR) определяет формат принимаемых и передаваемых кадров. Бит SB функционирует даже при запрещении подачи тактового сигнала UART. В обычном состоянии на выводе TX

13 Контроллер порта UART

выдаётся высокий уровень; когда UART-порт не задействуется, этот вывод может использоваться как выходной флаг.

Регистр управления линией связи UART (UART_LCR)

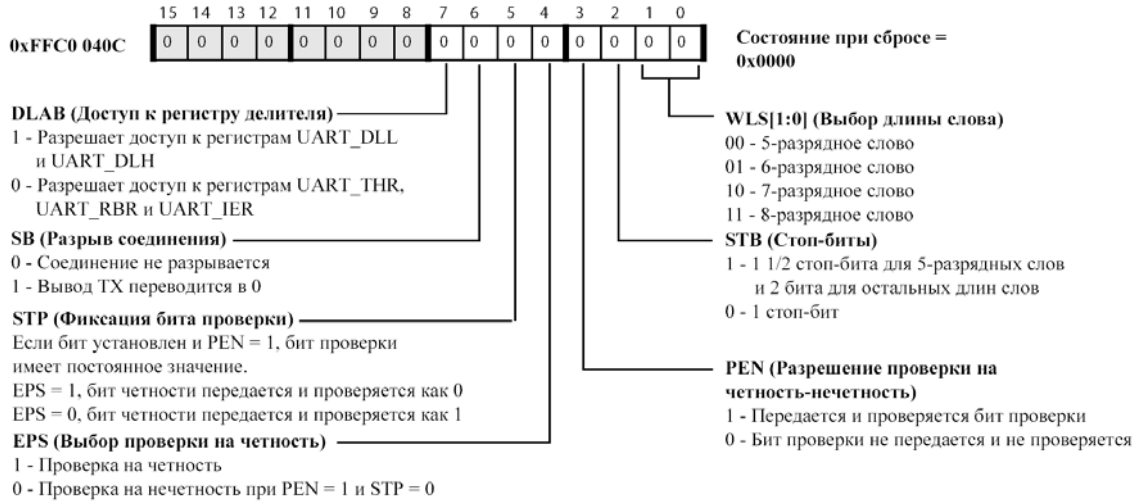


Рис. 13-2. Регистр управления линией связи UART

Регистр UART_MCR

Регистр управления модемом (UART_MCR), показанный на рис. 13-3, управляет работой порта UART. Даже если функции модема не поддерживаются, регистр управления модемом доступен для поддержки режима обратной связи.

Регистр управления модемом (UART_MCR)

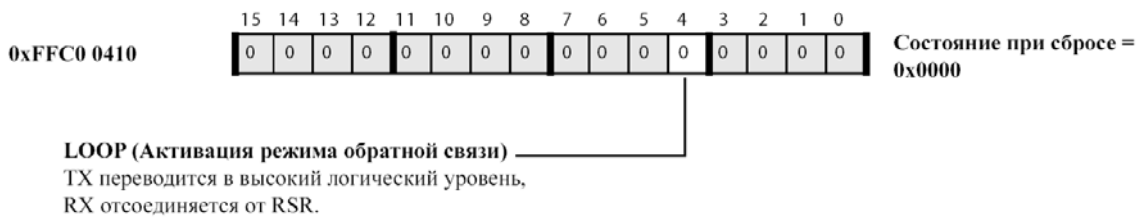


Рис. 13-3. Регистр управления модемом UART

В режиме обратной связи вывод TX переводится в состояние высокого уровня, и вход приёмника отсоединяется от вывода RX. При этом принимаемые данные внутренне перенаправляются на выход передатчика.

Регистр UART_LSR

Регистр состояния линии связи UART (UART_LSR), показанный на рис. 13-4, содержит информацию состояния порта UART.

Регистр состояния линии связи UART (UART_LSR)

Только для чтения

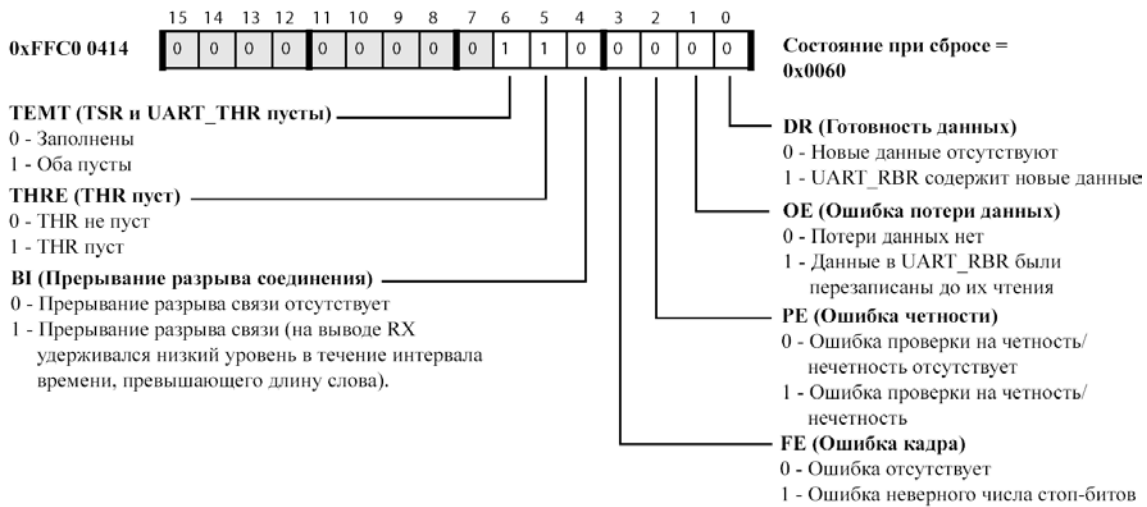


Рис. 13-4. Регистр состояния линии связи UART

Биты прерывания разрыва связи (VI), ошибки переполнения (OE), ошибки чётности (PE) и ошибки кадра (FE) сбрасываются при чтении регистра состояния линии связи UART (UART_LSR). Бит готовности данных (DR) сбрасывается при чтении регистра буфера приёма UART (UART_RBR).



Так как при чтении данных регистров определённые биты сбрасываются, при выполнении этих операций следует соблюдать осторожность. Дополнительную информацию см. в разделах главы 6 “Спекулятивное выполнение загрузки” и “Условное выполнение загрузки”.

Бит THRE указывает, что канал передачи UART готов к передаче новых данных, и программа может выполнить запись в UART_THR. При записи в UART_THR бит THRE сбрасывается, и снова устанавливается при копировании данных из UART_THR в сдвиговый регистр передачи (TSR). Бит TEMT может использоваться для проверки завершения начатой ранее операции передачи.

Регистр UART_THR

Запись в регистр буфера передачи UART (UART_THR) инициирует операцию передачи. Данные пересылаются во внутренний сдвиговый регистр передачи (TSR), в котором они сдвигаются на выход со скоростью, равной $SCLK / (16 \times \text{делитель})$, с добавлением необходимых старт-битов, стоп-битов и битов чётности. Передача слова данных всегда начинается со старт-бита (перехода из 1 в 0). При пересылке данных из UART_THR в сдвиговый регистр передачи в регистре состояния линии связи UART (UART_LSR) устанавливается флаг состояния THRE (Регистр буфера передачи пуст).

Адрес в карте памяти регистра UART_THR, доступного только для записи, совпадает с адресом регистра UART_RBR, доступного только для чтения, и

13 Контроллер порта UART

регистра `UART_DLL`. Для доступа к `UART_THR` должен быть сброшен бит `DLAB` в регистре `UART_LCR`. Когда бит `DLAB` сброшен, при записи по данному адресу изменяется содержимое регистра `UART_THR`, при чтении – возвращается содержимое регистра `UART_RBR`.

Необходимо отметить, что данные передаются и принимаются, начиная с младшего бита (бит 0).

Регистр буфера передачи UART (`UART_THR`)

Только для записи

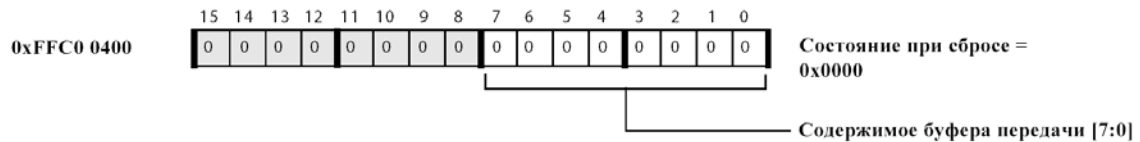


Рис. 13-5. Регистр буфера передачи UART

Регистр `UART_RBR`

При приёме используется тот же формат данных, что и при передаче, за исключением числа стоп-битов, которое всегда равно 1. После обнаружения стартового бита, принимаемое слово сдвигается в сдвиговый регистр приёма со скоростью, равной $SCLK / (16 \times \text{делитель})$. После приёма соответствующего числа битов (включая стоп-бит) данные и информация состояния обновляются, и содержимое сдвигового регистра приёма передаётся в регистр буфера приёма UART (`UART_RBR`), показанный на рис. 13-6. После передачи принятого слова в буфер `UART_RBR` и задержки, требуемой для синхронизации, обновляется флаг состояния `DR` (Готовность данных).

Частота выборки битов в 16 раз превышает скорость передачи; выборка выполняется как можно ближе к середине бита. Так как внутренняя частота выборки не всегда в точности соответствует скорости асинхронного приёма данных, точка выборки может быть смещена относительно центра каждого бита. Точка выборки синхронизируется по стартовому биту каждого слова; таким образом, ошибка накапливается только в течение приёма одного слова. Импульсы, длительность которых в два раза меньше длительности импульса тактовой синхронизации, удаляются приёмным фильтром.

Адрес в карте памяти регистра `UART_RBR`, доступного только для чтения, совпадает с адресом регистра `UART_THR`, доступного только для записи, и регистра `UART_DLL`. Для доступа к `UART_RBR` должен быть сброшен бит `DLAB` в регистре `UART_LCR`. Когда бит `DLAB` сброшен, при записи по данному адресу изменяется содержимое регистра `UART_THR`, при чтении – возвращается содержимое регистра `UART_RBR`.

Регистр буфера приема UART (UART_RBR)

Только для чтения

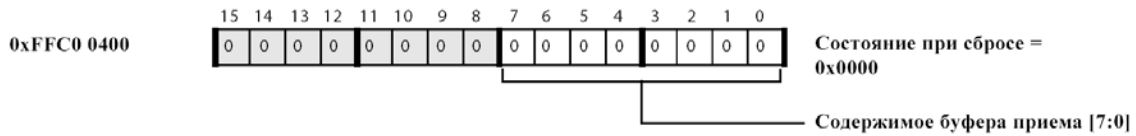


Рис. 13-6. Регистр буфера приёма UART

Регистр UART_IER

Регистр разрешения прерываний UART (UART_IER) используется для разрешения запросов прерываний, сигнализирующих системе о том, что регистры данных UART порта заполнены или пусты. Обычно, если для управления приёмом/передачей не используется опрос битов состояния, в этом регистре устанавливаются биты ERBFI и/или ETBEI.

Если данный регистр настраивается определённым образом, и DMA не используется, UART сигнализирует процессору о приёме/передаче данных при помощи прерываний. Для правильной работы в данном режиме необходимо разрешить генерацию прерываний системы и настроить соответствующие программы обработки прерываний. Для совместимости с предыдущими версиями в данном режиме регистр UART_IIR отображает достоверное состояние прерываний.

- ❗ Порт UART имеет три отдельных канала прерываний для независимой обработки приёма данных, передачи данных и состояния линии связи, независимо от того, разрешён ли DMA.

Когда DMA разрешён, обмен данными между процессором и портом UART выполняются в режиме DMA. Для выполнения приёма/передачи данных имеется два выделенных канала DMA. Обработка ошибок линии связи может настраиваться полностью независимо от настройки операций приёма/передачи.

Адрес в карте памяти регистра UART_IER совпадает с адресом регистра UART_DLH. Для доступа к UART_IER должен быть сброшен бит DLAB в регистре UART_LCR.

13 Контроллер порта UART

Регистр разрешения прерываний UART (UART_IER)

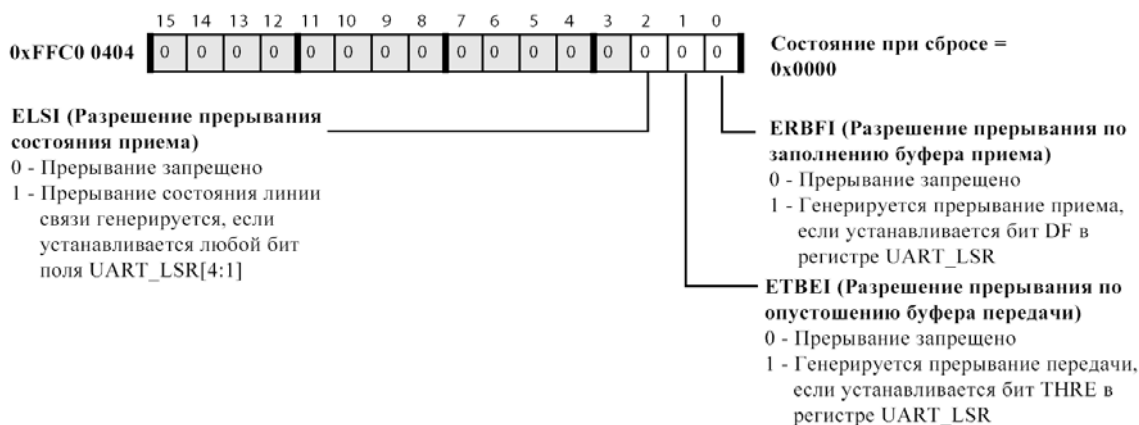


Рис. 13-7. Регистр разрешения прерываний UART

Для разрешения работы порта UART в режиме DMA необходимо настроить регистры управления DMA системы и затем разрешить прерывания, установив бит ERBFI и/или ETBEI в регистре UART_IER. Это необходимо, так как линии запроса прерываний также используются в качестве линий запроса DMA. В зависимости от того, разрешён DMA или нет, блок управления DMA либо выполняет операцию прямого доступа к памяти либо передаёт прерывание UART в блок обработки прерываний системы. Прерывание ошибки UART поступает в блок обработки прерываний системы напрямую, в обход контроллера DMA.

Установка бита ELSI разрешает генерацию прерывания в независимом канале прерывания, когда выполняется любое из следующих условий (устанавливается соответствующий бит в регистре состояния линии связи UART (UART_LSR)):

- ошибка потери данных при приёме (OE);
- ошибка чётности при приёме (PE);
- ошибка кадра при приёме (FE);
- прерывание разрыва связи (BI)

Когда в регистре UART_IER устанавливается бит ETBEI, модуль UART незамедлительно генерирует прерывание или формирует запрос DMA. При инициации передачи строки специальная обработка первого символа не требуется. Для этого следует установить бит ETBEI, загрузить в программе обслуживания прерывания первый символ из памяти и записать его в регистр UART_THR, как при передаче любого другого символа. Аналогично, при завершении передачи строки следует сбросить бит ETBEI.

Регистр UART_IIR

Для совместимости с предыдущими версиями, в процессоре имеется регистр идентификации прерываний UART (UART_IIR), отражающий состояние прерываний порта UART. В программах написанных для предыдущих версий процессора, может применяться назначение всем источникам прерываний UART

единого канала прерывания и совместное обслуживание прерываний одной программой обслуживания. Это может быть достигнуто назначением всем прерываниям порта UART одинакового приоритета в контроллере прерываний системы (SIC).

Когда бит ожидания обслуживания прерывания (NINT) сброшен, прерывание ожидает обслуживания. Поле STATUS указывает ожидающее обслуживания прерывание с наибольшим приоритетом. Высший приоритет имеет прерывание состояния линии приёма, прерывание опустошения UART_THR имеет низший приоритет. В случае одновременного возникновения обоих прерываний при чтении регистра UART_IIR возвращается значение 0x06.

Когда прерывание UART активно, программа обслуживания прерывания должна явно сбросить запрос прерывания. На следующем рисунке показано, как сбросить любой из трёх запросов прерывания.

Запрос прерывания передачи сбрасывается при записи новых данных в регистр UART_THR или при чтении регистра UART_IIR. Таким образом, если в программе не требуется дальнейшая передача данных, можно выполнить чтение регистра UART_IIR.

Регистр идентификации прерываний UART (UART_IIR)

Только для чтения



Рис. 13-8. Регистр идентификации прерываний UART

Если программа останавливает передачу данных, в ней должно выполняться чтение регистра UART_IIR, сбрасывающее запрос прерывания. Пока при чтении регистра UART_IIR возвращается значение 0x04 или 0x06 (сигнализирующее, что другое прерывание высокого приоритета ожидает обслуживания), запрос прерывания опустошения UART_THR не будет сброшен.



Если в SIC прерыванию состояния линии связи или прерыванию приёма назначается низкий приоритет, может возникнуть тупиковая ситуация. Чтобы избежать этого, всегда назначайте прерыванию опустошения UART_THR низший приоритет среди разрешённых прерываний UART.

13 Контроллер порта UART

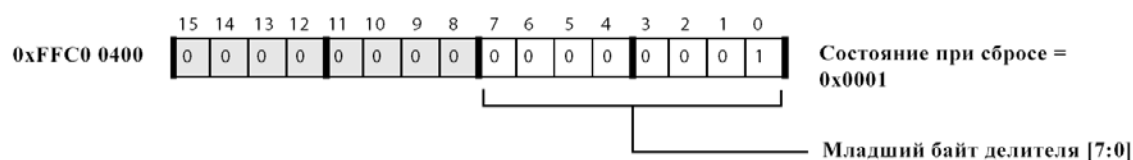
i Так как при чтении регистра UART_IIR сбрасываются запросы прерывания, при выполнении этой операции следует соблюдать осторожность. Дополнительную информацию см. в разделах главы 6 “Спекулятивное выполнение загрузки” и “Условное выполнение загрузки”.

Регистры UART_DLL и UART_DLH

Скорость передачи порта UART определяется тактовой частотой системы и 16-разрядным делителем. Значение 16-разрядного делителя задаётся регистрами младшего байта (UART_DLL) и старшего байта (UART_DLH) делителя UART. Для получения значения скорости передачи частота тактовой синхронизации UART делится на 16:

$$\begin{aligned} \text{Скорость передачи} &= \text{SCLK} / (16 \times \text{делитель}) \\ \text{делитель} &= 65536 \text{ при } \text{UART_DLL} = \text{UART_DLH} = 0 \end{aligned}$$

Регистр младшего байта делителя UART (UART_DLL)



Регистр старшего байта делителя UART (UART_DLH)

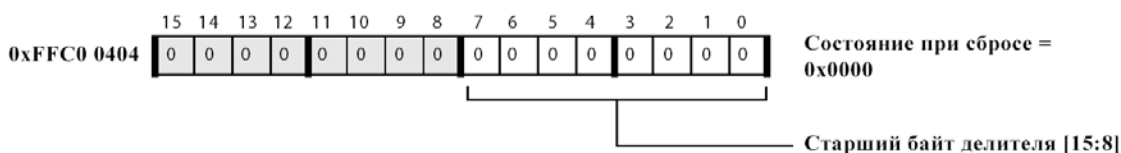


Рис. 13-9. Регистры делителя UART

Адрес регистра UART_DLL в карте памяти совпадает с адресом регистров UART_THR и UART_RBR. Адрес регистра UART_DLH в карте памяти совпадает с адресом регистра разрешения прерываний (UART_IER). Для обращения к регистрам делителя UART необходимо предварительно установить бит DLAB в регистре UART_LCR.

i Следует отметить, что при сбросе значение 16-разрядного делителя, формируемое регистрами UART_DLH и UART_DLL устанавливается в 0x0001, задавая по умолчанию максимально возможную тактовую частоту. Если UART не используется, запрещение подачи тактового сигнала UART позволяет снизить потребление мощности. Запись в регистры UART_DLH и UART_DLL может выполняться до или после установки бита UCEN.

В таблице 13-1 перечислены коэффициенты деления, которые необходимо задать для установки большинства стандартных значений скоростей передачи.

Таблица 13-1. Примеры скоростей передачи порта UART при частоте SCLK = 100 МГц.

Скорость передачи	DL	Действительное значение скорости передачи	Ошибка, %
2400	2604	2400.15	.006
4800	1302	4800.31	.007
9600	651	9600.61	.006
19200	326	19171.78	.147
38400	163	38343.56	.147
57600	109	57339.45	.452
115200	54	115740.74	.469
921600	7	892857.14	3.119
6250000	1	6250000	-



Выбор значений частоты SCLK, кратных требуемой скорости передачи, уменьшает процентное значение погрешности.

Регистр UART_SCR

Содержимое 8-разрядного регистра вспомогательных данных UART (UART_SCR) после сброса равно 0x00. Он используется для хранения данных общего назначения и не влияет на аппаратную часть порта UART.

Регистр вспомогательных данных UART (UART_GCTL)

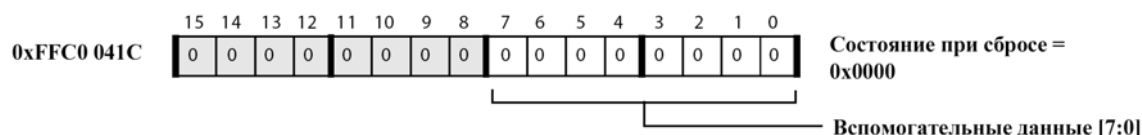


Рис. 13-10. Регистр вспомогательных данных UART

Регистр UART_GCTL

Регистр глобального управления UART (UART_GCTL) содержит биты разрешения подачи внутренних тактовых сигналов UART и биты разрешения работы порта UART в режиме IrDA.

Следует отметить, что в реализациях порта UART в предыдущих версиях процессоров бит UCEN отсутствовал. Он предназначен для снижения потребляемой мощности в случае, когда UART не используется. При переносе кода необходимо позаботиться об установке данного бита.

Биты изменения полярности приёма IrDA и изменения полярности передачи IrDA имеют значение только при работе в режиме IrDA. Два бита принудительной

13 Контроллер порта UART

генерации ошибки, FPE и FFE, предназначены для использования в программах отладки, особенно в режиме обратной связи.

Регистр глобального управления UART (UART_GCTL)

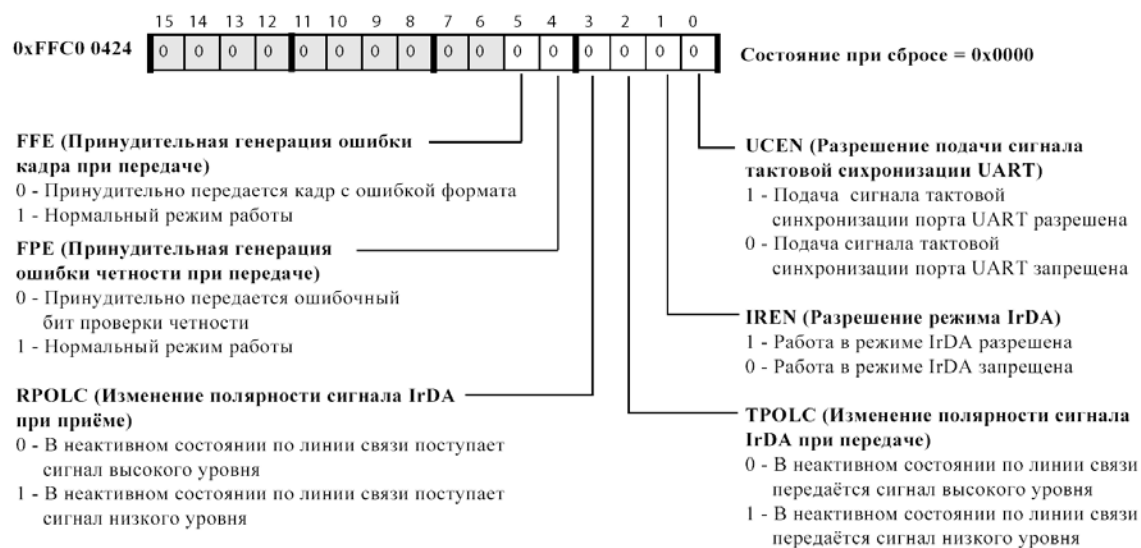


Рис. 13-11. Регистр глобального управления UART

Режим передач без DMA

В режиме передач без DMA передача/приём порта UART управляется ядром процессора. Для передачи символа выполняется загрузка регистра UART_THR. Принимаемые данные могут быть прочитаны в регистре UART_RBR. Процессор должен выполнять запись и чтение только одного символа за один раз.

Для предотвращения потерь данных и ошибок выравнивания последовательного потока данных используется квитирование при помощи двух флагов состояния в регистре состояния линии связи UART (UART_LSR) – THRE и DR.

Флаг THRE устанавливается, когда регистр UART_THR готов к приёму новых данных, и сбрасывается, когда процессор загружает в него новые данные. Запись в UART_THR при наличии в нём данных приводит к обновлению содержимого регистра, при этом предыдущий символ передан не будет.

Флаг DR сигнализирует о наличии новых данных в регистре UART_RBR. Этот флаг автоматически сбрасывается, когда процессор выполняет чтение регистра UART_RBR. Если чтение регистра UART_RBR происходит, когда он не полностью заполнен, возвращается предыдущее принятое значение. Если не выполняется своевременное чтение UART_RBR, его содержимое перезаписывается новыми данными, и устанавливается флаг потери данных (OE).

Когда прерывания запрещены, возможен опрос флагов состояния для определения готовности данных к пересылке. Следует отметить, что опрос флагов требует выполнения процессором большого количества операций и, поэтому, обычно не используется в задачах обработки сигналов в реальном времени. До разрешения подачи сигнала тактовой синхронизации порта UART программа может выполнить запись до двух слов в регистр UART_THR. Эти два слова передаются сразу после установки бита UCEN.

Альтернативным методом работы с портом UART является запись и чтение данных в программах обслуживания прерываний. Порт UART имеет отдельные линии прерываний приёма, передачи и ошибки. Генерация этих прерываний независимо друг от друга разрешается в регистре UART_IER.

Программы обслуживания прерываний могут определять источник прерываний на основании значения поля битов состояния в регистре идентификации прерываний UART (UART_IIR). Если прерывания поступают одновременно от нескольких источников, поле состояния указывает на прерывание с большим приоритетом. Для работы в этом режиме необходимо назначить соответствующие прерывания в контроллере прерываний процессора и снять маскирование. Программа обслуживания прерываний должна явно сбрасывать биты фиксации прерываний (см. рис. 13-8).

Передачи в режиме DMA

В данном режиме пересылки данных между портом UART и памятью выполняются по отдельным каналам DMA приёма (RX) и передачи (TX). В программе необходимо только настроить необходимые параметры передач либо в дескрипторном режиме, либо в режиме автобуферизации; далее передачи выполняются автоматически без вмешательства программы.

Так как в канале DMA UART отсутствует дополнительная буферизация данных, в нём возникают такие же задержки, как и в режиме передач без DMA. Однако они определяются активностью шины и механизмом арбитража, а не загруженностью процессора и приоритетами прерываний. Дополнительную информацию см. в главе 9, “Прямой доступ к памяти”.

В программах обслуживания прерываний DMA должна выполняться явная запись единиц в соответствующие регистры состояний IRQ DMA для сброса зафиксированных запросов прерываний.

Для разрешения работы порта UART в режиме DMA необходимо настроить регистры управления DMA системы и затем разрешить прерывания, установив бит ERBFI и/или ETBEI в регистре UART_IER. Это необходимо, так как линии запроса прерываний также используются в качестве линий запроса DMA. В зависимости от того, разрешён DMA или нет, блок управления DMA либо выполняет операцию прямого доступа к памяти либо передаёт прерывание UART в блок обработки прерываний системы. Прерывание ошибки UART поступает в блок обработки прерываний системы напрямую, в обход контроллера DMA.

При работе порта UART в режиме DMA поддерживаются 8-разрядные операции.

Смешанные режимы

В режимах DMA и программного ввода/вывода используются различные механизмы синхронизации. Следовательно, любые операции последовательной передачи данных должны быть завершены до перехода из режима работы без DMA в режим работы с DMA или наоборот. Другими словами, при переключении из режима передач без DMA в режим передач с DMA следует убедиться, что и `UART_THR` и сдвиговый регистр передачи (`TSR`) пусты при помощи проверки битов состояния `THRE` и `TEMT` в регистре `UART_LSR`. В случае переключения из режима передач с DMA в режим передач без DMA, процессор должен ожидать сброса 2-разрядного поля состояния буфера DMA в соответствующем регистре конфигурации канала DMA передачи UART (`UART_CONFIG_TX`).

При переключении из режима передач с DMA в режим передач без DMA необходимо убедиться в окончании пересылки данных и в канале DMA передачи (TX) и в канале DMA приёма (RX), включая данные в буферах FIFO DMA. Прерывание приёма в режиме DMA указывает, что последнее слово данных было записано в память (и покинуло FIFO DMA), прерывание передачи в режиме DMA указывает, что последнее слово данных было взято из памяти (и поступило в FIFO DMA). Для безопасного запрещения работы канала DMA процессор должен ждать, пока FIFO TX станет пуст, опрашивая бит состояния `DMA_RUN` в регистре `IRQ_STATUS` канала передачи. Этот бит сбрасывается при передаче последнего слова из FIFO TX.

Поддержка IrDA

Помимо стандартных функций UART порт UART процессора также поддерживает полудуплексную последовательную инфракрасную передачу данных, в соответствии с рекомендациями ассоциации разработчиков систем инфракрасной передачи данных (IrDA, Infrared Data Association). На физическом уровне, также известном как IrDA SIR (скорость передачи 9.6/115.2 Кбит/с), используется инверсная модуляция с возвратом к нулю (RZI, return-to-zero-inverted). Позиционно-импульсная модуляция не поддерживается.

RZI-модуляция достигается инвертированием и модуляцией кода без возврата к нулю (NRZ, non-return-to-zero), используемым в обычном режиме работы порта UART, при помощи тактового сигнала с частотой в 16 раз выше скорости передачи данных. На приёмной стороне сигнал с 16-кратным увеличением частоты используется для задания окна выборки импульса IrDA, используемого при восстановлении NRZ-кода из RZI-модулированного сигнала.

Поддержка IrDA разрешается установкой бита `IREN` в регистре глобального управления UART. При работе в режиме IrDA требуется применение внешних трансиверов.

Описание передатчика IrDA

Для генерации импульса IrDA, передаваемого по порту UART, NRZ-код, поступающий с выхода передатчика, сначала инвертируется (0 передаётся как импульс высокого логического уровня длительностью 16 периодов сигнала тактовой синхронизации UART, 1 передаётся как импульс низкого логического уровня длительностью 16 периодов сигнала тактовой синхронизации UART). Затем передний фронт импульса задерживается на 6 периодов сигнала тактовой синхронизации UART. Аналогичным образом, задний фронт импульса усекается на восемь периодов сигнала тактовой синхронизации UART. В конечном итоге 0 представляется в виде импульса высокого логического уровня длительностью 3/16 периода тактового сигнала, равного 16 периодам сигнала тактовой синхронизации UART. Импульс центрирован относительно середины битового интервала, как показано на рис. 13-12. Полученный IrDA импульс подаётся на внешний формирователь инфракрасного сигнала.

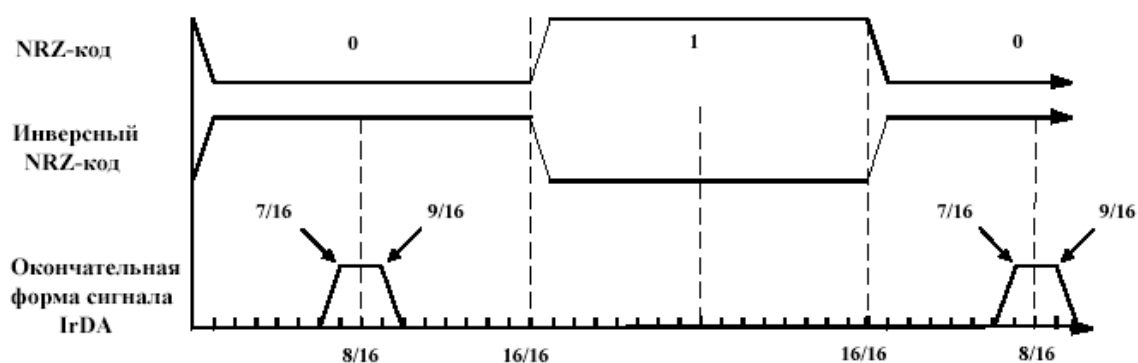


Рис. 13-12. Передаваемый импульс IrDA

При использовании данного подхода гарантируется выдача портом UART импульса высокого логического уровня длительностью три такта в каждые 16 тактов сигнала синхронизации UART. Как показано в таблице 13-1 ошибки, вносимые генератором сигнала синхронизации, очень малы и удовлетворяют погрешностям, определённым в большинстве спецификаций инфракрасных трансиверов.

Описание приёмника IrDA

Работа приёмника IrDA более сложна, чем работа передатчика. Приёмник должен обеспечивать выделение IrDA-импульса и устранять влияние шумов. Для этого приёмник выполняет поиск IrDA-импульса в узком окне, центрированном, относительно середины ожидаемого импульса.

Фильтрация кратковременных импульсных помех достигается за счёт отсчёта 16 тактов тактового сигнала системы от момента появления импульса. Если импульс заканчивается до истечения счётчика, принимается решение о наличии импульсной помехи. В противном случае принимается решение о наличии нуля. Этот метод приемлем, так как импульсы, возникающие вследствие внешних

13 Контроллер порта UART

емкостных перекрёстных связей, делятся не более доли периода тактового сигнала системы. Внешние источники помех могут быть исключены при помощи экранирования. При этом единственным источником импульсных помех будет являться передатчик. Надёжность работы порта UART процессора в режиме IrDA зависит от передатчика. Если параметры передатчика не соответствуют спецификации, могут возникнуть непредсказуемые результаты. 4-разрядный счётчик вносит дополнительный уровень защиты при минимальных аппаратных затратах. Так как частота тактового сигнала системы может изменяться в различных системах, следует отметить, что наибольшая допустимая длительность импульсной помехи обратно пропорциональна частоте тактового сигнала системы.

Окно выборки при приёме определяется счётчиком, который тактируется сигналом с частотой в 16 раз выше частоты сигнала выборки битов. Ресинхронизация окна выборки выполняется по каждому старт-биту путём центрирования окна на интервале его приёма.

Полярность данных при приёме выбирается при помощи бита IRPOL. На рис. 13-13 приведены примеры сигналов для каждого типа полярности.

- При $IRPOL = 0$ предполагается, что неактивное состояние входных данных при приёме – 0, активный переход в 1 соответствует значению 0 NRZ-кода в обычном режиме работы порта UART.
- При $IRPOL = 1$ предполагается, что неактивное состояние входных данных при приёме –1, активный переход в 0 соответствует значению 0 NRZ-кода в обычном режиме работы порта UART.

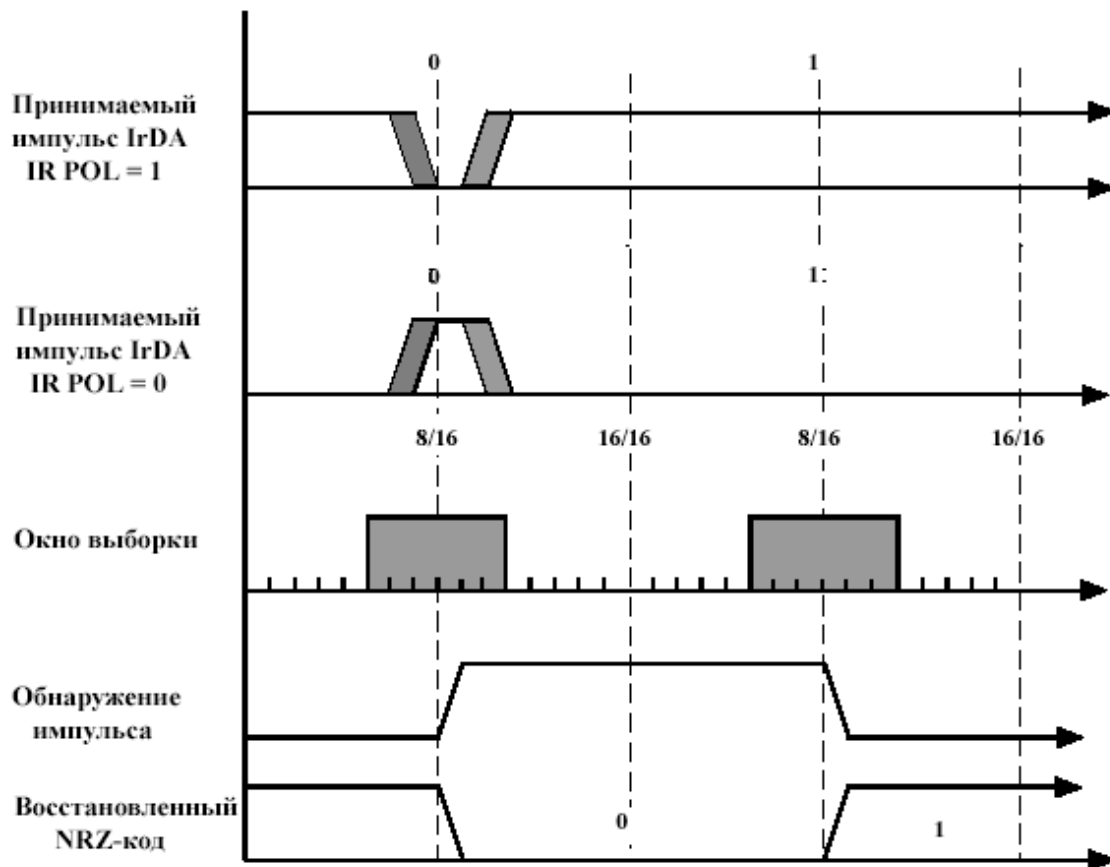


Рис. 13-13. Обнаружение импульса приёмником в режиме IrDA