

Регистровый состав и основные принципы работы с модулем USART микроконтроллера PIC18F458

| | |
|---------------------------------------|---|
| Регистровый состав модуля USART | 1 |
| Инициализация модуля USART | 2 |
| Настройка скорости передачи..... | 3 |
| Передача байтов..... | 4 |
| Приём байтов..... | 4 |
| Прерывания модуля USART | 5 |

Примечание: в данной статье описаны настройка модуля USART, а также приём и передача сообщений для наиболее простейшего случая: используем 8-битный режим и неприоритетные прерывания от приёмного буфера. Работа модуля происходит в асинхронном высокоскоростном режиме на скорости 115200 бод, тактовая частота микроконтроллера равна 20 МГц.

Регистровый состав модуля USART

Встроенный модуль USART микроконтроллера PIC18F458 состоит из нескольких частей: приёмника, передатчика и генератора скорости. Каждый из этих элементов в свою очередь состоит из регистров и рабочей логики. Регистры используются для буферизации принимаемых и передаваемых сообщений, а также настройки скорости передачи и управления логикой модуля.

Структурные схемы передатчика и приёмника модуля приведены на **Рис. 1** и **2** соответственно.

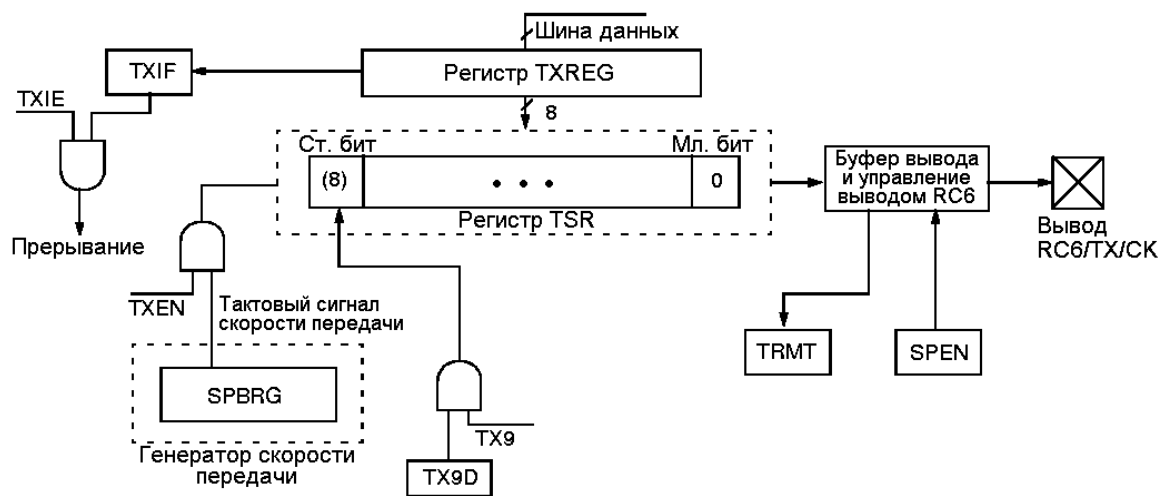


Рис. 1. Структурная схема приёмника модуля USART

Приёмник содержит следующие регистры:

RCSTA - регистр управления, содержит биты для настройки работы приёмника и флаги ошибок.

RCREG - двойной FIFO буфер для принимаемых данных.

RSR - сдвиговый регистр для приёма данных с линии, программно не доступен.

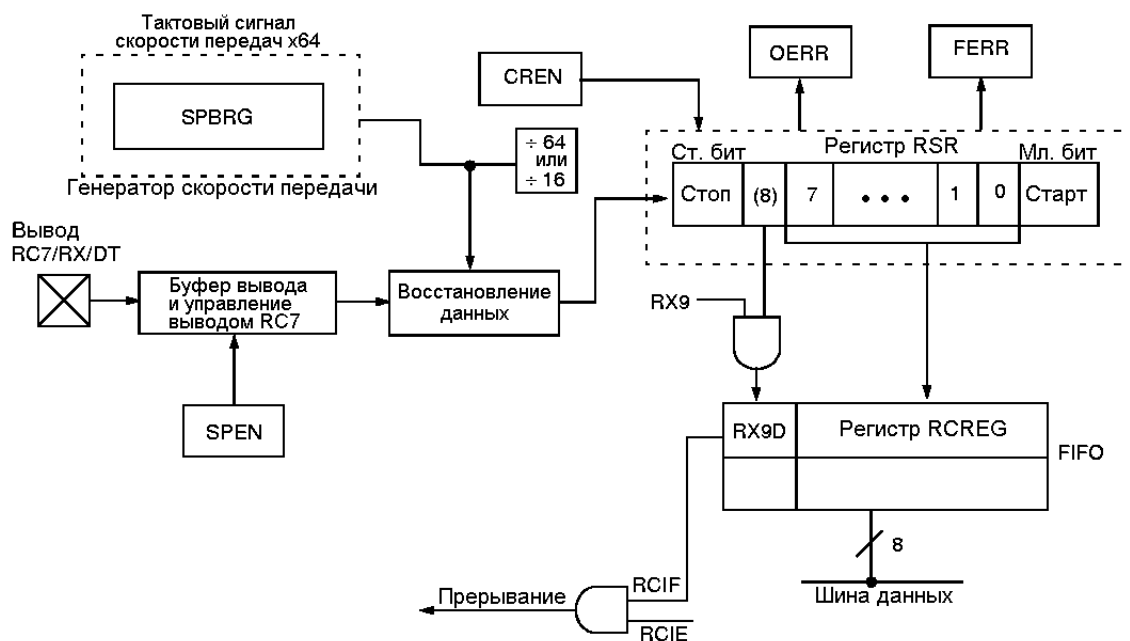


Рис. 2. Структурная схема передатчика модуля USART

Передатчик содержит следующие регистры:

TXSTA - регистр управления, содержит биты для управления работой передатчика.

TXREG - буфер для передаваемых данных.

TSR - сдвиговый регистр для передачи данных на линию, программно не доступен.

Генератор скорости содержит регистр **SPBRG**, в который заносится значение периода встроенного генератора обмена, а также с ним связан бит **BRGH** регистра **TXSTA**, управляющий скоростным режимом генератора.

Рассмотрим подробнее основные принципы работы с модулем USART.

Инициализация модуля USART

Перед тем как начать работу с модулем USART в режиме асинхронного последовательно интерфейса RS-232, необходимо выполнить его настройку. Для этого нужно выполнить следующие действия:

1) выходы **RC6** и **RC7** порта PORTC настроить как входы, для этого биты <7> и <6> регистра TRISC устанавливаем в единицу;

2) выполнить настройку скорости передачи данных, используя бит **BRGH** в регистре TXSTA для выбора скоростного режима и регистр **SPBRG**, в который необходимо поместить значение периода встроенного генератора скорости обмена. Более подробное описание настройки скорости приведено ниже;

3) настроить модуль USART на асинхронный режим работы сбросом в ноль бита **SYNC** в регистре TXSTA;

4) включить модуль USART установкой бита **SPEN** в регистре RCSTA.

5) запретить прерывание от передатчика модуля сбросом бита **TXIE** регистра PIE1;

6) отключить 9-битный приём и передачу сбросом в ноль битов **RX9** и **TX9** в регистрах RCSTA и TXSTA соответственно.

7) разрешить передачу и приём установкой битов **TXEN** и **CREN** в регистрах TXSTA и RCSTA соответственно.

8) разрешить прерывание от приёмника модуля установкой бита **RCIE** в регистре PIE1.

Приоритеты прерываний использовать не будем, поэтому соответствующие настройки оставляем без изменений.

Настройка скорости передачи

Модуль USART имеет встроенный генератор скорости обмена в бодах, период которого определяется значением в регистре SPBRG. В асинхронном режиме можно использовать бит выбора скоростного режима BRGH регистра TXSTA, который тоже влияет на скорость обмена. Этот бит определяет, в каком режиме работает генератор скорости обмена.

Генератор скорости обмена может работать в одном из двух режимов: высокоскоростном (BRGH = 1) и низкоскоростном (BRGH = 0). В зависимости от режима работы значение регистра SPBRG вычисляется по разным формулам. Эти формулы для расчета скорости обмена и значения регистра SPBRG в асинхронном режиме приведены в **Таблице 1**. Скорость обмена рассчитывается в зависимости от используемой тактовой частоты микроконтроллера.

Таблица 1. Формулы для расчёта скорости обмена и значения регистра SPBRG

| Значение бита BRGH | BRGH = 0 (низкоскоростной режим) | BRGH = 1 (высокоскоростной режим) |
|--------------------------------|---|---|
| Расчёт скорости обмена | $\text{Скорость обмена} = \frac{F_{osc}}{64 \cdot (X + 1)}$ | $\text{Скорость обмена} = \frac{F_{osc}}{16 \cdot (X + 1)}$ |
| Расчёт значения регистра SPBRG | $X = \frac{F_{osc}}{\text{Скорость обмена} \cdot 64} - 1$ | $X = \frac{F_{osc}}{\text{Скорость обмена} \cdot 16} - 1$ |

В этой таблице параметр X – значение регистра SPBRG для заданной скорости обмена, F_{osc} – тактовая частота.

Будем использовать высокоскоростной режим работы, так как формула $F_{osc}/16 \cdot (X + 1)$ позволяет уменьшить погрешность скорости обмена.

Погрешность скорости обмена рассчитывается по формуле:

$$\text{Погрешность} = 100\% \cdot (\text{Заданная скорость} - \text{Полученная скорость}) / \text{Заданная скорость} .$$

Заметим, что запись нового значения в регистр SPBRG сбрасывает таймер BRG, гарантируя немедленный переход на новую скорость.

Рассчитаем значение регистра SPBRG для скорости обмена 115200 бод для тактовой частоты микроконтроллера $F_{osc} = 20 \text{ МГц}$:

$$X = \frac{20\,000\,000}{115\,200 \cdot 16} - 1 = 9.85 .$$

Приведём к целому числу: $X = [9.85] = 10$ (берём наиболее близкое целое число – тогда погрешность будет меньше).

При таком значении будет получена скорость

$$\text{Скорость обмена} = \frac{20\,000\,000}{16 \cdot (10 + 1)} = 113\,636 \text{ бод} .$$

Полученная скорость не равна заданной скорости. Рассчитаем погрешность скорости:

$$\text{Погрешность} = 100\% \cdot (113\,636 - 115\,200) / 115\,200 = -1.36\% .$$

Это означает, что полученная скорость меньше заданной на 1.36 %, что будет приводить к ошибкам при передаче информации. В этом случае необходимо использовать проверку достоверности пакетов с помощью контрольной суммы и повторную передачу пакетов, которые оказались повреждёнными.

Таким образом, чтобы получить скорость, приблизительно равную заданному значению 115200 бод, в регистр SPBRG необходимо занести значение 10, а бит BRGH в регистре TXSTA установить в единицу.

Примечание: мы упоминали, что формула для высокоскоростного режима позволяет уменьшить погрешность. Давайте, для проверки сравним полученную погрешность в 1.36% с той, которая получится для формулы низкоскоростного режима $F_{osc}/64 \cdot (X + 1)$:

$$X = \frac{20\,000\,000}{115\,200 \cdot 64} - 1 = 1.71.$$

Приведём к целому числу: $X = [1.71] = 2$.

$$\text{Скорость обмена} = \frac{20\,000\,000}{64 \cdot (2 + 1)} = 104\,167 \text{ бод.}$$

$$\text{Погрешность} = 100\% \cdot (104\,167 - 115\,200) / 115\,200 = -9.58 \%$$

Как мы видим – погрешность в этом случае действительно больше. Поэтому, если имеется возможность выбрать высокоскоростной режим, то лучше использовать его – это позволит уменьшить количество ошибок.

Передача байтов

За передачу байтов в асинхронном режиме отвечает асинхронный передатчик модуля. Для передачи байта в нём реализованы следующие регистры:

TXREG - передающий буфер, в который программно загружается байт для передачи.

TSR - передающий сдвиговый регистр, который побитово выдаёт загруженный в него байт на вывод TX модуля. Этот регистр является программно недоступным. Байт загружается в TSR из регистра TXREG либо после того, как передан последний стоп-бит предыдущего байта, либо после установки бита TXEN регистра TXSTA, если он до этого был сброшен.

Передача разрешается установкой бита TXEN в регистре TXSTA в единицу.

После завершения передачи очередного байта устанавливается бит TXIF в регистре PIR1, который указывает на то, что регистр TXREG пуст и готов принять следующий байт. Этот бит сбрасывается аппаратно, когда в регистр TXREG заносится новое значение. Если в регистре PIE1 установлен бит TXIE (разрешены прерывания от модуля передачи USART), то при установке бита TXIF будет сгенерировано прерывание.

Таким образом, чтобы осуществить передачу данных, необходимо выполнить следующие действия:

- 1) установить в единицу бит TXEN в регистре TXSTA, чтобы разрешить передачу (включить передатчик);
- 2) ждать, пока бит TXIF в регистре PIR1 не установится в единицу, после чего перейти к следующему шагу;
- 3) как только бит TXIF установился в единицу (регистр TXGER свободен), загрузить в регистр TXREG очередной байт данных для передачи.

Шаги 2 и 3 выполняются до тех пор, пока не будут переданы все байты данных. Шаг 1 выполняется на этапе инициализации модуля.

Если необходимо прервать передачу, то бит TXEN нужно сбросить в ноль. Это вызовет немедленное прекращение передачи, сброс передатчика и перевод вывода TX в высокоимпедансное состояние.

Приём байтов

За приём байтов в асинхронном режиме отвечает асинхронный приёмник модуля. Для приёма байта в нём реализованы два регистра:

RSR – приёмный сдвиговый регистр, выполняет побитовый приём данных с вывода RX, преобразует их в байт и помещает в регистр RCREG. Программно не доступен.

RCREG – приёмный буфер, в который помещаются поступающие байты данных. Представляет собой двухуровневый FIFO-буфер и может принять два байта данных, которые считываются из него в порядке поступления, при этом в регистр RSR в тот же момент времени может приниматься третий байт. Данные считываются из регистра RCREG программно.

Приём разрешается установкой бита CREN регистра RCSTA в единицу.

После записи байта из регистра RSR в RCREG в регистре PIR1 устанавливается флажок **RCIF**, указывающий на то, что в приёмный буфер поступил новый байт данных. Если бит RCIE в регистре PIE1 установлен, то генерируется прерывание. Флаг RCIF сбрасывается аппаратно при чтении данных из регистра RCREG.

При приёме данных могут произойти ошибки переполнения и кадрирования. Первая ошибка происходит, если в регистре RCREG находятся два байта данных, и при этом обнаруживается стоп-бит третьего байта, принимаемого в RSR. Вторая ошибка происходит, если при приёме байта не обнаружен стоп-бит. Как правило, ошибка кадрирования происходит из-за рассинхронизации скоростей обменивающихся устройства, вызванной погрешностью при настройке скорости с помощью регистра SPBRG (см. **Настройка скорости передачи**).

Если произошла ошибка переполнения, устанавливается бит **OERR** в регистре RCSTA и логика приёмника отключается. Поэтому, чтобы продолжить приём данных, необходимо сбросить флаг ошибки переполнения. Сброс бита OERR выполняется отключением приёмника, то есть сбросом в ноль бита CREN в регистре RCSTA. Чтобы продолжить приём данных, после сброса бита CREN, его нужно установить в единицу. В этом случае может возникнуть необходимость предварительно очистить регистр RCREG от данных, иначе новый поступивший байт снова вызовет ошибку переполнения. Для очистки регистра RCREG нужно два раза прочесть из него данные.

Если произошла ошибка кадрирования, в регистре RCSTA устанавливается бит **FERR**. Этот бит буферизуется вместе с принятыми данными и при чтении данных из регистра RCREG он заменяется значением, связанным со следующим байтом данных. В случае если необходимо сохранить этот флажок вместе с байтом данных, то сначала следует считать значение регистра RCSTA, в котором находится этот флажок, и только затем – сами данные.

Таким образом, для приёма данных нужно выполнить следующие действия:

- 1) разрешить прерывание по принятию байта установкой в единицу бита RCIE в регистре PIE1;
- 2) разрешить приём (включить приёмник) установкой бита CREN регистра RCSTA в единицу;
- 3) если произошла ошибка переполнения, сбросить флаг ошибки OERR сбросом и установкой бита CREN;
- 4) если произошла ошибка кадрирования, считать данные из регистра RCREG;
- 5) прочесть байт из регистра RCREG в программный буфер.

Шаги 1 и 2 выполняются на этапе инициализации модуля. Шаги 3 – 5 выполняются в подпрограмме обработки прерывания от приёмника.

Прерывания модуля USART

Модуль USART может генерировать прерывания при передаче и по приёму байта данных. Здесь мы используем прерывание по приёму байта, чтобы определить момент, когда модуль получает с линии данных очередной байт. Прерывание от приёмника разрешается установкой бита RCIE в регистре PIE3. Как отмечалось выше, это прерывание генерируется при поступлении очередного байта данных в регистр RCREG. При этом устанавливается флажок RCIF. Это флажок доступен только для чтения и сбрасывается аппаратно при чтении данных из регистра RCREG.

Для модуля USART не выделено отдельного прерывания при возникновении ошибок, так как все ошибки (переполнения и формата кадра) происходят при приёме сообщения. Поэтому в подпрограмме обработки прерывания от приёмника кроме чтения данных из регистров RCREG и RCSTA нужно выполнять и все необходимые действия по обработке произошедших ошибок (см. **Приём байтов**).

Для прерываний можно задать приоритет, но если эта возможность не используется, как в нашем случае, её можно отключить сбросом бита IPEN в регистре RCON.

Следует добавить, что кроме разрешения прерываний от модуля USART необходимо установить бит глобального разрешения прерываний **GIE** и бит разрешения прерываний от периферийных модулей **PEIE** в регистре INTCON.