

Глава 3

Параметры COM и LPT портов компьютера

Практически каждый современный ПК оборудован хотя бы одним асинхронным последовательным COM-портом и, как правило, одним параллельным LPT-портом. Эти порты предназначены для подключения к компьютеру внешних устройств и аппаратов. Далее приводятся краткий набор необходимых сведений сначала по COM-портам, затем – по LPT-портам.

В конце этой главы книги вы найдете краткую информацию о USB-портах

Аппаратная реализация COM-порта

Обычно компьютер оснащен одним или двумя портами последовательной передачи данных. Эти порты расположены, как правило, на материнской плате компьютера. Каждому COM-порту соответствует несколько регистров, через которые компьютерная программа может управлять портом, и определенная линия прерываний IRQ для сигнализации компьютеру о состоянии порта.

Порт последовательной передачи данных может использоваться очень широко. К нему может быть подключен манипулятор мышь, модем, сканер, графопостроитель, принтер и т.п. Все эти аппараты и устройства через COM-порт проводят обмен данными с компьютером, используя при этом стандартные принципы последовательной передачи данных, заложенные в конструкции порта.

Последовательная передача данных означает, что данные передаются бит за битом по единственной линии связи. Формат передачи данных для передающего и приемного устройств должен быть один и тот же, иначе передача данных будет невозможна. Формат передачи данных определяют стартовый и стоповый биты, бит четности и скорость передачи.

Основу последовательного асинхронного адаптера, который может управлять сразу несколькими COM-портами, составляет микросхема универсального асинхронного приемопередатчика UART (Universal Asynchronous Receiver Transmitter). Встречаются несколько разновидностей этой микросхемы – Intel 8250, 16450, 16550, 16550A. При этом часть из них уже безнадежно устарела

Для каждого COM-порта указанные микросхемы содержат регистры передатчика и приемника данных, а также несколько управляющих регистров, доступных через команды ввода/вывода.

При передаче очередной байт записывается в буферный регистр передатчика, откуда затем переписывается в сдвиговый регистр. Затем байт «выдвигается» из сдвигового регистра по битам. Аналогично работают сдвиговый и буферный регистры приемника.

Программа имеет доступ только к буферным регистрам. Копирование информации в сдвиговые регистры и сдвиг данных выполняется микросхемой UART автоматически. Регистры, управляющие асинхронным последовательным портом, будут описаны ниже.

Внешне каждый COM-порт асинхронного последовательного адаптера представлен собственным разъемом. Существует два стандарта на разъемы COM-порта: DB25 и DB9. Первый разъем имеет 25, а второй 9 выводов. Несмотря на то, что разъем DB25 содержит

в два с половиной раза больше выводов, чем DB9, они передают одинаковые сигналы. При необходимости можно приобрести переходник между разъемами DB25 и DB9.

В табл. 3.1 приведена разводка разъема DB25 со стороны последовательного асинхронного адаптера:

Таблица 3.1. Разводка разъема DB25

Номер контакта	Назначение контакта	Вход или выход
1	Защитное заземление (Frame Ground, FG)	—
2	Передаваемые данные (Transmitted Data, TD)	Выход
3	Принимаемые данные (Received Data, RD)	Вход
4	Запрос для передачи (Request to send, RTS)	Выход
5	Сброс для передачи (Clear to Send, CTS)	Вход
6	Готовность данных (Data Set Ready, DSR)	Вход
7	Сигнальное заземление (Signal Ground, SG)	—
8	Детектор принимаемого с линии сигнала (Data Carrier Detect, DCD). Иногда сигнал DCD обозначают как CD (Carrier Detect)	Вход
9-19	Не используются	
20	Готовность выходных данных (Data Terminal Ready, DTR)	Выход
21	Не используется	
22	Индикатор вызова (Ring Indicator, RI)	Вход
23-25	Не используется	

В табл. 3.2 приведена разводка разъема DB9 со стороны последовательного асинхронного адаптера:

Таблица 3.2. Разводка разъема DB9

Номер контакта	Назначение контакта	Вход или выход
1	Детектор принимаемого с линии сигнала (Data Carrier Detect, DCD). Иногда сигнал DCD обозначают как CD.	Вход
2	Принимаемые данные (Received Data, RD)	Вход
3	Передаваемые данные (Transmitted Data, TD)	Выход
4	Готовность выходных данных (Data Terminal Ready, DTR)	Выход
5	Сигнальное заземление (Signal Ground, SG)	—
6	Готовность данных (Data Set Ready, DSR)	Вход
7	Запрос для передачи (Request to send, RTS)	Выход
8	Сброс для передачи (Clear to Send, CTS)	Вход
9	Индикатор вызова (Ring Indicator, RI)	Вход

Процесс подтверждения связи

В начале сеанса связи компьютер должен удостовериться, что модем может произвести вызов абонента (находится в рабочем состоянии). После вызова абонента модем должен сообщить компьютеру, что он соединился с удаленной системой. Подробнее это происходит следующим образом.

Компьютер устанавливает сигнал на линии DTR, чтобы показать модему, что он готов к проведению сеанса связи. В ответ модем устанавливает сигнал на линии DSR. Когда модем произвел соединение с другим удаленным модемом, он устанавливает сигнал на линии DCD, чтобы сообщить об этом компьютеру.

Падение напряжения на линии DTR сообщает модему, что компьютер не может продолжать сеанс связи, например из-за выключения питания компьютера. Тогда модем прервет связь. Падение напряжения на линии DCD сообщает компьютеру, что модем потерял связь и не может больше продолжать соединение с удаленным модемом.

Процесс управления потоком

В предыдущем разделе был рассмотрен процесс подтверждения связи между компьютером и модемом, точно такой же процесс будет происходить и между двумя компьютерами. Далее рассмотрим механизм, с помощью которого можно регулировать передачу данных от компьютера к модему и наоборот.

Когда одно устройство (например, компьютер), пытается передать данные с большей скоростью, чем они могут быть обработаны принимающей системой (модемом), результатом может стать потеря части передаваемых данных. Чтобы предотвратить передачу большего числа данных, используют управление связью, называемое *управление потоком*.

Стандарт RS-232-C определяет возможность управления потоком только для *полудуплексного соединения*, при котором в каждый момент времени данные могут передаваться только в одну сторону.

Однако фактически этот механизм используется и для *дуплексных соединений*, когда данные передаются по линии связи одновременно в двух направлениях.

В полудуплексных соединениях компьютер подает сигнал RTS, когда ему надо передать данные. Модем отвечает сигналом по линии CTS, когда он готов, и компьютер начинает передачу данных. До тех пор пока оба сигнала RTS и CTS не примут активное состояние, только модем может передавать данные.

При дуплексных соединениях сигналы RTS и CTS имеют противоположные значения по сравнению с теми, которые они имели для полудуплексных соединений.

Когда компьютер может принять данные, он подает сигнал по линии RTS. Если при этом модем готов для принятия данных, он возвращает сигнал CTS. Если напряжение на линии RTS или CTS падает, то это сообщает передающей системе, что получающая система не готова для приема данных.

Соединение двух компьютеров. Нуль-модем

При соединении по COM-портам двух терминальных устройств (двух компьютеров) как минимум необходимо перекрестное соединение линий TD и RD.

Однако в большинстве случаев этого недостаточно, так как для устройств DTE (компьютер) и DCE (модем) функции, выполняемые линиями DSR, DTR, DCD, CTS и RTS, асимметричны.

Устройство DTE подает сигнал DTR и ожидает получения сигналов DSR и DCD. В свою очередь, устройство DCE подает сигналы DSR, DCD и ожидает получения сигнала DTR. Таким образом, если вы соедините вместе два устройства DTE кабелем, который вы использовали для соединения устройств DTE и DCE, то они не смогут договориться друг с другом. Не выполнится процесс подтверждения связи.

Рассмотрим действия сигналов RTS и CTS, которые управляют потоком данных. Иногда для соединения двух устройств DTE эти линии соединяют вместе на каждом конце кабеля. В результате получаем, что другое устройство всегда готово для получения данных. При большой скорости передачи принимающее устройство не всегда успевает принимать и обрабатывать данные, возможна потеря данных.

Чтобы решить все эти проблемы, для соединения двух устройств типа DTE () используется специальный кабель, в обиходе называемый «нуль-модем».

Используя два разъема типа DB25 и кусок кабеля, можно легко спаять нуль-модемный кабель самостоятельно, руководствуясь схемами, изображенными на рис. 3.2.

Нуль-модемный кабель, представленный на схеме, расположенной слева на рис. 3.2, содержит значительно меньше проводов, чем нуль-модемный кабель, изображенный на схеме справа. Это получается за счет того, что на каждом конце кабеля линии RTS и CTS, а также и линии DSR, DCD и DTR соединены вместе. При этом процедуры подтверждения связи и управления потоком всегда будут заканчиваться успешно. На больших скоростях это может привести к потере информации, поэтому мы рекомендуем использовать вторую схему, расположенную справа. Изображенные на рис. 3.2 схемы во всех случаях используют разъемы типа DB25.



Рис. 3.2. Схема кабеля «нуль-модем»

Очень часто может возникнуть такая ситуация, что из-за наличия двух типов разъемов (DB25 и DB9) бывают нужны переходники с одного типа разъемов на другой. Например, вы можете использовать такой переходник для соединения COM-порта компьютера и кабеля нуль-модема, если на компьютере установлен разъем DB25, а кабель оканчивается разъемом DB9.

Схема такого переходника приведена на рис. 3.3. Слева на схеме располагается разъем DB25, а справа – разъем DB9.

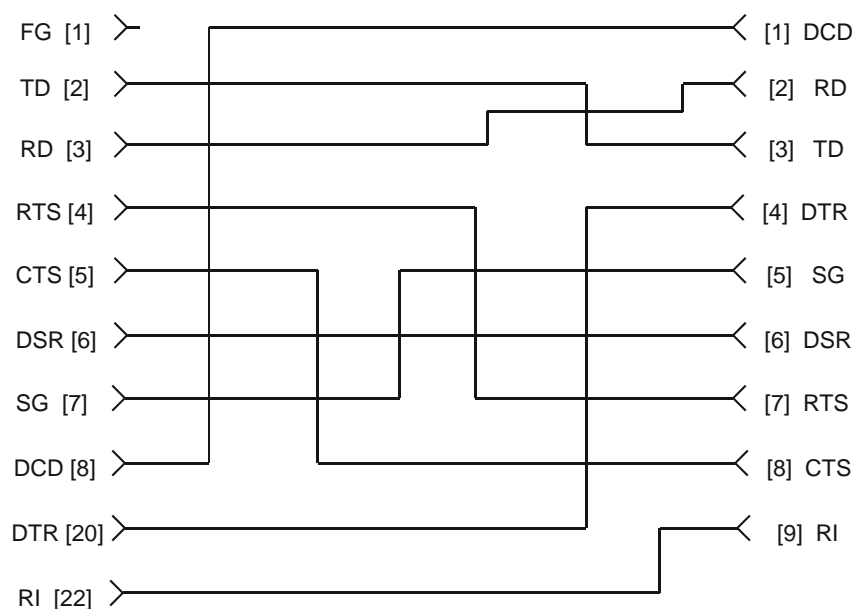


Рис. 3.3. Схема переходника для разъемов DB25 и DB9

Изображенная на рис. 3.4 схема представляет собой так называемую «заглушку для COM-порта», которая применяется при проверке COM-портов компьютера специальными тестирующими программами. Эту схему можно также применять при отладке собственного коммуникационного программного обеспечения.

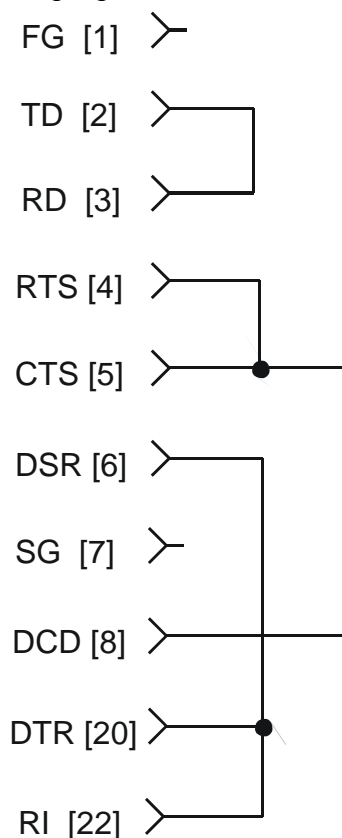


Рис. 3.4. Схема «заглушки» для COM-порта

Порты асинхронного адаптера

При запуске в работу операционной системы компьютера модуль начальной загрузки BIOS тестирует имеющиеся порты RS-232-C и инициализирует их. В зависимости от версии BIOS инициализирует первые два или четыре порта. Их базовые адреса записываются в области данных BIOS начиная с адреса 0040:000h.

Адреса COM-портов из нашего примера являются своего рода стандартом. Первый адаптер COM1 обычно имеет базовый адрес 3F8h и занимает диапазон адресов от 3F8h до 3FFh. Второй адаптер COM2 имеет базовый адрес 2F8h и занимает адреса 2F8h...2FFh. Третий адаптер COM3 имеет базовый адрес 3F8h и занимает диапазон адресов от 3E8h до 3EFh. Четвертый адаптер COM4 имеет базовый адрес 2E8h и занимает адреса 2E8h...2EFh. Тем не менее для некоторых компьютеров, например с шиной MCA-PS/2, адреса COM-портов могут иметь другие значения.

Порты асинхронного адаптера могут вырабатывать прерывания:

- COM1, COM3 – IRQ4 (соответствует INT 0Ch;
- COM2, COM4 – IRQ3 (соответствует INT 0Bh).

Порты COM1 и COM3 асинхронного последовательного адаптера используют линию IRQ4 и вырабатывают прерывание INT 0Ch, а порты COM2 и COM4 используют линию IRQ3 и вырабатывают прерывание INT 0Bh.

Некоторые платы последовательного асинхронного адаптера позволяют присвоить COM-портам другие линии IRQ, например IRQ5 или IRQ7.

Как видите, порты COM1, COM3 и COM2, COM4 используют одинаковые прерывания. Поэтому, как правило, нельзя одновременно использовать порты COM1 и COM3, так же как порты COM2 и COM4. Например, если к порту COM1 подключен модем, то мышь можно подключить только к порту COM2 или COM4.

Каждый из COM-портов имеет в своем составе восемь отдельных портов, называемых *регистрами*. При этом в каждом регистре имеется восемь ячеек, в которые записывается команда, которая представляет собой шестнадцатеричное число. В программах командное число зачастую записывается, для наглядности, в двоичном виде, при этом каждой ячейке регистра соответствует один элемент командного числа, выраженного в двоичном виде.

Номера ячеек регистра располагаются справа налево, в соответствии с записью чисел в двоичном формате. На рис. 3.5 схематично изображено расположение ячеек регистра.

D7	D6	D5	D4	D3	D2	D1	D0
-----------	-----------	-----------	-----------	-----------	-----------	-----------	-----------

Рис. 3.5. Схема расположения ячеек регистра

Далее перейдем к подробному рассмотрению регистров (портов), используемых COM-портами.

1. Регистр данных

Регистр данных расположен непосредственно по базовому адресу COM-порта и используется для обмена данными и для задания скорости обмена. Например, для COM1 адрес регистра данных будет 3F8h (т.е. $base_adr = 3F8h$).

При передаче данных в этот регистр необходимо записать *передаваемый байт данных*. После приема данных от внешнего устройства принятый байт можно также прочитать из этого же регистра.

В зависимости от состояния старшего бита управляющего регистра (расположенного по адресу $base_adr + 3$, где $base_adr$ соответствует базовому адресу COM-порта) назначение этого регистра может изменяться. Если старший бит равен нулю, регистр используется для записи передаваемых данных. Если же старший бит равен единице, регистр используется для ввода значения младшего байта делителя частоты тактового генератора. Изменяя содержимое делителя, можно изменять скорость передачи данных. Старший байт делителя записывается в регистр управления прерываниями по адресу $base_adr + 1$.

Зависимость скорости передачи данных от значения делителя частоты представлена в следующей табл. 3.3:

Таблица 3.3. Зависимость скорости от делителя частоты

Делитель , десятичная форма	Делитель , шестнадцатеричная форма	Скорость передачи , бит/с
1040	600h	110
768	300h	150
384	180h	300
192	0C0h	600
96	60h	1200
48	30h	2400
24	18h	4800
12	0Ch	9600
8	8h	14400
6	6h	19200
3	3h	38400
2	2h	57600
1	1h	115200

Как следует из этой таблицы, максимальная скорость обмена информацией, которую можно достичь при использовании асинхронного адаптера, достигает 115200 бит/с, что примерно соответствует 12 Кбайт в секунду.

2. Регистр управления прерываниями

Этот регистр имеет адрес $base_adr + 1$ (для COM1 это будет $3F8h + 1 = 3F9h$) и используется либо для управления прерываниями от асинхронного адаптера, либо (после вывода в управляющий регистр байта с установленным в единицу старшим битом) для вывода значения старшего байта делителя частоты тактового генератора.

В режиме управления прерываниями ячейки регистра выполняют следующие функции:

D0 – разрешает прерывания при готовности принимаемых данных. Если бит содержит единицу, генерация прерывания при готовности принимаемых данных разрешена.

D1 – разрешает прерывания после передачи байта (когда выходной буфер передачи пуст). Если бит содержит единицу, генерация прерывания после передачи байта разрешена.

D2 – разрешает прерывания по обнаружению состояния BREAK или по ошибке. Если бит содержит единицу, то при обнаружении состояния BREAK или при возникновении ошибки происходит прерывание.

D3 – разрешает прерывания по изменению состояния входных линий на разъеме RS-232-C (CTS, DSR, RI, DCD). Если бит содержит единицу, то при изменении состояния линий CTS, DSR, RI и DCD COM-порт вырабатывает прерывание.

D7-D4 – не используются, должны быть равны нулю.

3. Регистр идентификации прерывания

Этот регистр имеет адрес $base_adr + 2$ (для COM1 это будет $3F8h + 2 = 3FAh$).

После того как произошло прерывание от COM-порта, программа может прочитать значение регистра идентификации прерываний, чтобы определить причину его возникновения.

Формат регистра представлен ниже:

D0 – если бит равен единице, значит, нет прерываний, ожидающих обслуживания.

D2-D1 – содержит идентификатор прерывания, значения которого показаны в таб. 3.4.

Таблица 3.4. Причины прерывания

Значение бит D2-D1	Причина прерывания
00	Изменилось состояние модема. Устанавливается при изменении состояния входных линий CTS, RI, DCD, DSR. Сбрасывается после чтения состояния модема из регистра состояния модема.
01	Буфер передатчика пуст. Сбрасывается при записи новых данных в регистр данных
10	Данные приняты и доступны для чтения. Сбрасывается после чтения данных из регистра данных
11	Прерывание по линии состояния приемника; возникает при переполнении приемника, ошибках четности или формата данных или при состоянии BREAK. Сбрасывается после чтения состояния линии из регистра состояния линии

D7-D3 – эти ячейки должны быть равны нулю.

4. Управляющий регистр

Этот регистр имеет адрес $base_adr + 3$ (для COM1 это будет $3F8h + 3 = 3FBh$).

Управляющий регистр доступен для записи и чтения. Этот регистр управляет различными характеристиками UART:

□ скоростью передачи данных,

- ☐ контролем четности,
- ☐ передачей сигнала BREAK,
- ☐ длиной передаваемых слов (символов).

D1-D0 – данные в этих ячейках определяют длину передаваемых слов в битах. Зависимость длины слова от данных в ячейках показаны в табл. 3.5.

Таблица 3.5. Длина передаваемых слов

Значение бит D1- D2	Длина передаваемых слов, бит
00	5
01	6
10	7
11	8

D2 – значение этой ячейки определяет количество стоповых бит. Зависимость числа стоповых бит от содержания ячейки D2 показано в табл. 3.6.

Таблица 3.6. Число стоповых бит

Значение бита D2	Число стоповых бит
0	1
1	2

D3-D4 – значения этих ячеек управляют проверкой на четность, зависимость показана в табл. 3.7.

Таблица 3.7. Режим проверки четности

Значение бит D3-D4	Режим проверки четности
00,10	Контроль на четность не выполняется
01	Выполняется проверка на нечетность
11	Выполняется проверка на четность

D5 – влияет на фиксацию четности. При установке этого бита, бит четности всегда принимает значение 0, если биты **D3 D4** равны 11, или принимает значение 1, если биты **D3 D4** равны 01

D6 – влияет на установка перерыва. Вызывает вывод строки нулей в качестве сигнала BREAK для подключенного устройства

D7 – значение этой ячейки используется для доступа к регистру установки скорости. Если бит в **D7** равен единице, то регистр данных и регистр управления прерываниями используются для загрузки делителя частоты тактового генератора. Если бит равен нулю, то регистр данных и регистр управления прерываниями используются как обычно

5. Регистр управления модемом

Этот регистр имеет адрес $base_adr + 4$ (для COM1 это будет $3F8h + 4 = 3FCh$).

Регистр управления модемом управляет состоянием выходных линий DTR, RTS и линий OUT1 и OUT2, специфических для модемов.

Формат ячеек регистра представлен ниже:

D0 – состояние этой ячейки устанавливает значение линии DTR. Является сигналом подтверждения связи. Используется модемами для разрешения передачи данных между компьютером и микросхемой UART

D1 – устанавливает состояние линии RTS. Является сигналом подтверждения связи. Используется модемами для разрешения передачи данных между компьютером и микросхемой UART

D2 – устанавливает состояние линии OUT1. Для некоторых модемов при установке этого бита в единицу происходит его аппаратный сброс

D3 – устанавливает состояние линии OUT2. Если бит **D3** содержит единицу, то UART может вырабатывать прерывания, а если нуль – не может

D4 – выполняет запуск диагностики при входе асинхронного адаптера, замкнутом на его выход (Digital Loopback test – проверка при замкнутой петле). Эта возможность реализована только для асинхронных портов, использующих микросхему UART 8250, или полностью совместимых с ней

D7-D5 – значения этих ячеек должны быть равны нулю

6. Регистр состояния линии

Этот регистр имеет адрес *base_adr* + 5 (для COM1 это будет 3F8h + 5 = 3FDh).

Регистр состояния линии позволяет программе определить причину ошибок, которые могут произойти при передаче данных компьютером и COM-портом. Формат ячеек регистра представлен следующим образом.

D0 – значение 1 в ячейке показывает, что данные получены и готовы для чтения, при чтении данных бит сбрасывается

D1 – значение 1 сообщает об ошибке переполнения. Принят новый байт данных, а предыдущий еще не был считан программой. В результате предыдущий байт всегда бывает потерян.

D2 – значение 1 сообщает об ошибке четности, сбрасывается после чтения состояния линии

D3 – значение 1 сообщает об ошибке синхронизации. Возникает, например, при отсутствии стоп-битов в принятом байте

D4 – значение 1 сообщает об обнаружении запроса на прерывание передачи BREAK – длинная строка нулей

D5 – значение 1 сообщает о том, что регистр хранения передатчика пуст. Можно записывать новый байт для передачи

D6 – значение 1 сообщает о том, что регистр сдвига передатчика пуст. Этот регистр получает данные из регистра хранения и преобразует их в последовательный вид для передачи. Если этот бит равен единице, то UART может принять очередной символ от компьютера.

D7 – значение 1 сообщает о тайм-ауте (устройство не связано с компьютером)

7. Регистр состояния модема

Этот регистр имеет адрес *base_adr* + 6 (для COM1 это будет 3F8h + 6 = 3FEh).

Регистр состояния модема позволяет программе определить состояние управляющих сигналов, передаваемых модемом асинхронному порту компьютера. Формат регистра состояния модема представлен в следующих строках.

D0 – если бит **D0** равен единице, линия CTS изменила состояние

D1 – если бит равен единице, линия DSR изменила состояние

D2 – если бит **D2** равен единице, линия R1 изменила состояние. Некоторые коммуникационные программы определяют по состоянию этого бита наличие звонка на телефонной линии

D3 – если данный бит равен единице, значит, линия DCD изменила свое состояние. Некоторые коммуникационные программы определяют по состоянию этого бита, установил ли модем соединение с удаленным модемом

D4 – бит соответствует состоянию линии CRS. Эта линия используется совместно с линией RTS при реализации аппаратного управления потоком данных

D5 – бит соответствует состоянию линии DSR. Эта линия используется совместно с линией DTR при аппаратной реализации подтверждения связи

D6 – бит соответствует состоянию линии R1. Единица означает, что модем обнаружил звонок на телефонной линии

D7 – бит определяет состояние линии DCD. Единица означает, что модемом получена несущая частота. Заметим, что при выполнении аналогового теста (analog test) этот бит должен содержать единицу. Если это не так, то возможно, что модем исправен (для внешних модемов), но кабель, соединяющий модем и компьютер, не полностью соответствует стандарту RS-232C.

8. Регистр вспомогательный

Этот регистр имеет адрес *base_adr* + 7 (для COM1 это будет 3F8h + 7 = 3Fh).

Регистр не несет никакой функциональной нагрузки и может использоваться как буфер для хранения одного байта. В некоторых исполнениях RS-232C этот регистр может отсутствовать.

Технические параметры интерфейса RS-232-C

При передаче данных на большие расстояния из-за помех, наводимых электромагнитными полями, возможно возникновение ошибок. Вследствие этого накладываются ограничения на длину соединительного кабеля между устройствами DTE-DTE и DTE-DCE.

Официальное ограничение по длине для соединительного кабеля по стандарту RS-232-C составляет 15,24 метра. Однако на практике это расстояние может быть значительно больше. Оно непосредственно зависит от скорости передачи данных.

Уровни напряжения на линиях разъема составляют для логического нуля от -15 до -3 вольт, для логической единицы от +3 до +15 вольт. Промежуток от -3 до +3 вольт соответствует неопределенному значению.

При подключении внешнего устройства к разъему интерфейса RS-232-C (а также при соединении двух компьютеров нуль-модемом), и в компьютере и в подключаемом устройстве должно быть отключено питание. При этом компьютер должен иметь подсоединенное заземление. В противном случае можно вывести из строя асинхронный адаптер. Земля компьютера и земля внешнего устройства должны быть соединены вместе.

Параллельный порт (LPT)

Такое название обычно носит порт параллельного обмена данными, который иногда носит название «Параллельный интерфейс CENTRONICS».

Область применения

Этот интерфейс обеспечивает возможность бит-параллельного, байт-последовательного обмена информацией.

Линии интерфейса

Линии DATA состоят из восьми ячеек DATA 1...DATA 8. Предназначены для передачи информационных данных на подключаемое внешнее устройство (ПУ). Направление сигнала к ПУ.

Эти сигналы содержат информацию битов 1...8. Уровень «high» соответствует логической 1, а уровень «low» - логическому 0.

Линия -STROBE (-STROBE) (Направление сигнала к ПУ)

Уровень «high» на линии STROBE означает, что комбинация сигналов на информационных линиях недостоверна. Уровень «low» означает достоверность. Импульс «low» используется для записи данных в логику блока приема.

Линия -ACK (-ACK) (Направление сигнала от ПУ)

Импульс «low» на линии ACK показывает, что блок приема записал данные и готов к обработке дальнейших данных.

Линия PE (PE) (Направление сигнала от ПУ). Применяется для управления принтера.

Уровень «high» на линии PE показывает, что в ПУ подошла к концу бумага.

Линия BUSY (BUSY) (Направление сигнала от ПУ).

Уровень «high» на линии BUSY означает, что ПУ не может принимать данные. BUSY становится «high» в следующих случаях:

1. во время приема и обработки данных;
2. в состоянии OFF-LINE;
3. в состоянии «ошибка ПУ».

Линия SLCT (SELECT) (Направление сигнала от ПУ)

Уровень «low» на линии SELECT показывает, что ПУ не готово к приему. В этом состоянии линия BUSY тоже имеет уровень «high».

Линия -AUTOFEED_XT (-AUTO FEED XT) (Направление сигнала к ПУ).

Когда на этой линии присутствует импульс «low», блок приема вновь инициализируется так же, как при подключении напряжения питания.

Линия -INIT (-INIT) (Направление сигнала к ПУ).

Когда на линии INIT присутствует уровень «low», блок приема вновь инициализируется так же, как при подключении напряжения питания.

Линия -ERROR (-ERROR) (Направление сигнала от ПУ).

Уровень «low» на линии ERROR означает, что ПУ находится в одном из следующих состояниях:

1. PAPER END;
2. OFF LINE;

3. ERROR.

Линия -SLCT_IN (-SELECT IN) (Направление сигнала к ПУ).

Блок приема может записать комбинации сигналов на информационных линиях только тогда, когда на этой линии уровень “low”. При наличии уровня “high” обмен данными хотя и выполняется надлежащим образом, но ПУ игнорирует данные.

Линия Масса (GND)

Логический массовый уровень ПУ (заземление корпуса).

Электрические условия

Блок передачи: все выходы усилителей – стандартные логические элементы ТТЛ, усиливающие только одну линию и внутренне не используемые дополнительно.

Кабель интерфейса

Надлежит использовать скрученный парами кабель, е.е. кабель, где каждый сигнальный провод скручен с проводом массы. Провода должны иметь сечение не менее 0,08 мм и волновое сопротивление 60-150 Ом. Пучок кабеля должен быть экранирован для обеспечения показателей радиопомех. Экран надлежит с обеих сторон соединить с защитным проводом. При соблюдении электрических условий работоспособность интерфейса обеспечивается для длины линий не более 1,8 м.

Схема контактов разъема LPT

Схема включения 25-контактной гнездовой колодки разъема LPT-порта показана в табл. 3.8. Знак ‘-’ (минус) перед наименованием сигнала говорит о том, что сигнал является инвертируемым.

Таблица 3.8. Контакты разъема LPT

Контакт	Сигнал	Контакт	Сигнал
1	-STROBE	10	-ACK
2	DATA 1	11	BUSY
3	DATA 2	12	PE
4	DATA 3	13	SLCT
5	DATA 4	14	-AUTO FD
6	DATA 5	15	-ERROR
7	DATA 6	16	-INIT
8	DATA 7	17	-SLCT IN
9	DATA 8	18...25	Ground

Программирование параллельного интерфейса

Каждый параллельный порт имеет свое имя, например, LPT1, и свой адаптер, который может управляться тремя регистрами:

- ☐ регистром входных данных;
- ☐ регистром статуса (обычно это статус принтера);

□ регистром управления (обычно имеется в виду управление принтером).

Порт регистра выходных данных для LPT1 – 0x378.

Порт регистра статуса для LPT1 – 0x378 + 1

Биты

7	6	5	4	3	2	1	0
---	---	---	---	---	---	---	---

Выводы: 11 10 13 15

Бит 0 – если установлена 1, означает ошибку превышения времени, в течение которого принтер находится в состоянии занятости;

Бит 1 и 2 – не используются;

Бит 3 – если 0, то ошибка принтера;

Бит 4 – если 1, то принтер связан с ПК;

Бит 5 – если 1, то нет бумаги в принтере;

Бит 6 – если 0, то принтер подтверждает прием сигнала;

Бит 7 – если 0, то принтер занят.

Регистр управления устанавливает канал в исходное состояние и координирует вывод данных.

Порт регистра управления для LPT1 – 0x378 + 2

Биты

7	6	5	4	3	2	1	0
---	---	---	---	---	---	---	---

Бит 0 – кратковременное единичное значение воспринимается как стробирующий сигнал для вывода байта;

Бит 1 – если 1, то автоматический перевод строки после возврата каретки;

Бит 2 – если 0, то происходит инициализация порта и принтера;

Бит 3 – если 1, то принтер подключается к линии (режим online);

Бит 4 – если 1, то разрешается аппаратное прерывание от принтера;

Биты 5, 6 и 7 не используются.

Коротко о универсальной последовательной шине (USB)

Если сегодня практически все периферийные устройства подключаются к компьютеру через последовательный - COM и параллельный - LPT порты, то в ближайшее время эти порты исчезнут из стандартной конфигурации. Их неизбежно вытеснит универсальная последовательная шина – USB (Universal Serial Bus).

Краткий обзор шины USB

Универсальная последовательная шина (USB) была разработана в 1996 году для расширения архитектуры PC. Шина позволяет обеспечить такие основные требования как: высокую скорость передачи, одновременное подключение большого количества устройств, удобство операций, надежную связь, а также возможность горячего подключения устройств и низкую цену.

USB имеет топологию многоярусной звезды, где в центре находится главная ЭВМ, в средних узлах ретрансляторы (хабы), а в конечных точках индивидуальные устройства (функции). USB позволяет подключать к шине до 127 функций и дерево может быть глубиной до 6 уровней.

Главная ЭВМ всегда является ведущей и все обмены данными осуществляются под ее непосредственным управлением.

Передачи осуществляются в одном из двух режимов: с полным быстродействием – 12 Мбит/с и низкой скоростью – 1,5 Мбит/с. Полное быстродействие позволяет передавать данные в приложениях, работающих с интенсивными звуковыми сигналами или сжатым видео, а передачи с низкой скоростью поддерживают приложения с небольшой интенсивностью данных.

Протоколы передачи

Для передачи данных определены четыре типа обмена данными (транзакции).

- Изохронные транзакции обеспечивают пересылку данных в реальном масштабе времени, когда фиксированное количество данных передается за равные интервалы времени с фиксированной скоростью. Этот тип транзакций обеспечивает своевременность доставки данных, но их точностью не гарантируется, так как нет времени для повторной передачи пакетов с ошибками, которые просто игнорируются.
- Объемные транзакции обеспечивают точность данных, а своевременность – нет.
- Транзакции прерывания предназначены для связи с устройствами, которые имеют умеренные требования по скорости передачи данных. Главная ЭВМ периодически, через фиксированные интервалы времени, опрашивает эти устройства для того, чтобы определить имеются ли данные для передачи. В таких устройствах появление данных имеет спонтанный характер, но передача их должна осуществляться немедленно.
- Транзакции управления используются для конфигурирования при подключении и управления в процессе работы.
- Все транзакции по USB состоят из трех пакетов, каждый из которых начинается по инициативе главной ЭВМ.
- Пакет маркер описывает тип и направление передачи, адрес устройства USB и номер оконечной точки.
- Пакет данных, который передает источник, если у него есть данные.
- Пакет взаимодействия передается после успешного приема данных или источником, если у него нет данных для передачи. При обнаружении ошибок пакет взаимодействия не передается.

Для обнаружения ошибок каждый пакет имеет контрольные поля CRC-кодов, позволяющие обнаруживать все одиночные и двойные ошибки.

Кадры

Информационные сообщения на шине сгруппированы в кадры. Каждый кадр имеет длительность 1 мс и состоит из множественных передач. Каждый кадр начинается с передачи главной ЭВМ пакета маркера начала кадра, содержащего номер кадра. Внутри кадра транзакции могут быть повторены несколько раз.

Питание

Кроме сигнальной пары кабель USB имеет линии передачи напряжения питания 5 В. Устройства, подключенные к USB, могут иметь как собственный источник питания. Так питаться непосредственно от шины. При питании непосредственно от шины устройство может потреблять до 100 мА. Если для нормальной работы устройству требуется больший ток (до 500 мА), то оно должно запросить его у главной ЭВМ. Если главная ЭВМ определяет, что в настоящее время шина загружена и не может выделить больше тока, то устройство не может переключиться в режим большего потребления и должно остаться в конфигурации малого потребления.

Оконечные точки

Самым низким уровнем каждого устройства является функция, которых может быть одна или больше. Оконечная точка подразумевается как виртуальный порт для управления функцией. Каждая оконечная точка может быть источником или приемником данных.

В каждом устройстве для операций конфигурирования и общего управления должна быть реализована оконечная точка с номером 0. Устройства с полным быстродействием могут иметь до 15 доступных для использования оконечных точек, а с низкой скоростью только 2. Ввод и вывода данных на шине осуществляется относительно главной ЭВМ, а не устройства.

Нумерация устройств

USB поддерживает подключение и отключение устройств в процессе работы. Нумерация устройств на шине является постоянным процессом, отслеживающим динамические изменения физической топологии. Процедура нумерации осуществляется без вмешательства пользователя или прикладного программного обеспечения. Процедура нумерации осуществляется следующим образом.

- Когда устройство подключается к шине USB, главная ЭВМ определяет изменение состояния своего порта, а устройство переходит в режим «подключено».
- Главная ЭВМ формирует сигнал сброса на шине, после чего устройство переходит в состояние «включено», все его регистры устанавливаются в исходное состояние и оно отзывается на обращения по нулевому адресу.
- Главная ЭВМ сообщает устройству уникальный адрес и оно переходит в состояние «адресовано».
- Главная ЭВМ считывает все конфигурации устройства и, исходя из считанной информации, конфигурирует все имеющиеся оконечные точки данного устройства, устройство переходит в состояние «сконфигурировано» и становится готовым для использования.

Описатели

При подключении устройства к шине оно передает главной ЭВМ свои параметры, позволяющие идентифицировать устройство, определить оконечные точки и функции для каждой конечной точки. Существуют пять основных категорий описателей.

- **Описатель устройства** содержит общую информацию: фирма-производитель, номер изделия, серийный номер, число поддерживаемых конфигураций.
- **Описатель конфигурации** содержит информацию о потребляемой мощности устройства и количестве интерфейсов, поддерживаемых в этой конфигурации. Устройство может иметь более одной конфигурации (то есть, устройство с большим потреблением может также поддерживать конфигурацию при малом потреблении).
- **Описатель интерфейса** содержит число оконечных точек, используемых в этом интерфейсе, а также класс драйвера для поддержки устройства.
- **Описатель конечной точки** детализирует фактические регистры устройства. Содержит информацию о поддерживаемых типах передачи, направлении обмена (ввод/вывод), требуемой полосы пропускания канала и интервалах обслуживания. Может быть более одной оконечной точки в устройстве и оконечные точки могут быть разделены между различными интерфейсами.
- **Строковые описатели** используются для того, чтобы обеспечить пользователя специфической или специальной для приложения информацией. Они могут быть необязательными и закодированными в формате «Unicode». Перечисленные выше четыре описателя могут быть ссылкой или индексом к строковым описателям.

Классы устройств / драйверы класса

Операционные системы главной ЭВМ обычно имеют драйверы, которые группируют функции по общим признакам устройств, называемым классами. Например, класс может включать устройства хранения, звуковые, связи и пользовательского интерфейса, но не ограничиваться перечисленным. Ссылки на драйверы класса для данного устройства могут быть в двух описателях: устройства и интерфейса. Для большинства устройств может быть подобран стандартный драйвер для данного класса, который поддерживает большинство функций или команд. Если устройство не имеет стандартного драйвера для данного класса. То оно должно поставляться вместе со специфическим файлом >>.inf>> или драйвером для его поддержки.

Краткий обзор USB не может полностью раскрыть все тонкости интерфейса, поэтому дополнительно хотелось бы отметить, что передачи с низкой скоростью разработаны для устройств, которые в прошлом использовали прерывание для связи с главной ЭВМ. В системах с USB эти устройства, если имеют данные для передачи, не прерывают процессор непосредственно. А главная ЭВМ периодически опрашивает каждое устройство, для того чтобы определить, имеются ли у него данные для передачи. Скорость опроса между устройством и главной ЭВМ задается при конфигурировании и обеспечивает допустимое время ожидания обслуживания.

Для более подробного знакомства с USB обратитесь к спецификации USB VI.I, доступной в Интернете на сайте по адресу <http://www.usb.org>

И еще о USB

Протокол обмена данными по USB сложен и многие фирмы проводят необходимые разработки, к которым привлечено большое число специалистов. В то время, когда большинство фирм не раскрывают своих достижений в области работ по USB, английская компания [FTDI \(Future Technology Devices International\)](http://www.ftdichip.com) постоянно информирует общественность о своих достижениях, в том числе и в Интернете на сайте по адресу <http://www.ftdichip.com>.

Возможностью подключения к компьютеру USB-устройств сейчас никого не удивишь. По сравнению с традиционными портами ввода-вывода компьютера (последовательный - COM, параллельный - LPT) универсальная последовательная шина (Universal Serial Bus) обеспечивает более высокую скорость обмена данными. Максимальная пропускная способность USB версии 1.1 – 12 Мбит/с, более современной версии 2.0 – 480 Мбит/с. Для низкоскоростных устройств предусмотрена скорость 1,5 Мбит/с.

Соединять компьютер с периферийными устройствами с помощью USB очень просто. ***Разрешено состыковывать и растыковывать соединители***, не выключая компьютер, предусмотрено автоматическое распознавание устройства немедленно после его подключения с последующей установкой необходимых драйверов.

Область применения USB не ограничена мультимедийными приложениями. Этот скоростной, рассчитанный на обслуживание большого числа устройств интерфейс удобен для аппаратуры связи, сбора и хранения информации, которую традиционно подключают к портам COM и LPT компьютеров. К сожалению, у компьютерных пользователей всех стран мира имеется большое количество различных устройств, предназначенных для работы с компьютером через COM и LPT порты. Замена интерфейса в существующем устройстве довольно сложна. Один из способов решения проблемы - применение преобразователей различных интерфейсов в USB. На российском рынке уже появляются подобные устройства на основе микросхем английской компании FTDI (Future Technology Devices International). В настоящее время компания выпускает три многофункциональные микросхемы: FT8U100AX, FT8U232AM и FT8U245AM.

FT8U232AM - преобразователь USB в традиционный последовательный интерфейс - можно устанавливать в USB-модемах, переходниках COM-USB, сканерах штрих-кода, измерительной аппаратуре - фактически в любых устройствах, ранее использовавших сравнительно медленные интерфейсы RS-232, RS-422, RS-485. Она способна передавать данные в обе стороны со скоростью до 2000 кбит/с, причем пользователю не требуется никаких знаний об устройстве и работе USB. Поставляемые компанией FTDI программные драйверы создают впечатление, что обмен идет через обычный COM-порт.

Однако протокол обмена данными по USB сложен и реализовать его до недавнего времени было не под силу не только радиолюбителям, но и многим специалистам. Сегодня, установив в разрабатываемом приборе микросхему FT8U232AM или FT8U245AM, можно преобразовать USB в "виртуальный" последовательный или параллельный порт, обмен данными с которым ведут привычными хорошо известными методами.

Универсальная последовательная шина предназначена для подключения на высоких скоростях периферийных устройств к компьютеру. Количество устройств, одновременно соединенных через шину данных, может достигать 127.

USB работает по принципу временного разделения пакетов. Инициатором каждого сеанса связи может быть только компьютер. Длина кабеля при высокоскоростном режиме не должна превышать 5 метров, на низкой скорости — 3 метров. Это ограничение отсекает все попытки осуществить проводную связь между двумя компьютерами на большом расстоянии.

Для подключения устройств применяется четырехжильный кабель: два провода — питание +5 В и «земля»; два других — сигнальные. Вид сигнала NRZI (Non Return to Zero Invert to ones). Обычно кабель защищен экранирующей оплеткой и ферритовыми кольцами для фильтрации высокочастотных помех.

Фирма FTDI предлагает в своей линейке продуктов для USB многофункциональную микросхему FT8U100AX, на базе которой можно разработать концентратор. Эта микросхема позволяет подключить к USB-шине до 7 устройств. Также в FT8U100AX интегрированы интерфейсные блоки клавиатуры и мыши (PS/2), RS-232, IrDA SIR, IrDA CIR.

Подробные схемы и рекомендации по этому вопросу можно найти в Application Notes, выложенных на сайте FTDI в формате PDF.

Программное обеспечение для FT8U100AX также можно свободно скачать с официального сайта FTDI <http://www.ftdichip.com>. На сайте постоянно появляются обновленные версии драйверов. Возможно, в скором времени появится аналогичный кристалл, но уже совместимый с USB 2.0.

Микросхема производится в единственном типе корпуса 100-PQFP и работает в коммерческом температурном диапазоне — от 0 до +70 °C.

Кроме FT8U100AX компания FTDI предлагает микросхемы FT8U232AM(BM) и FT8U245AM(BM). Первая микросхема реализует конвертер последовательного интерфейса RS-232 в USB, вторая — буфер FIFO, позволяющий создавать устройства на базе произвольного микроконтроллера с поддержкой USB.

В нашей стране также проводятся разработки по применению USB. Одним из таких разработчиков является Институт радиотехники в г. Екатеринбург, который публикует большое количество информации о своих работах в Интернете на сайте, расположенном по адресу <http://www.institute-rt.ru/usb/>.

Информацию для радиолюбителей можно получить в Интернете на сайте по адресу <http://www.kirov.ru/~ra4nalr/>.