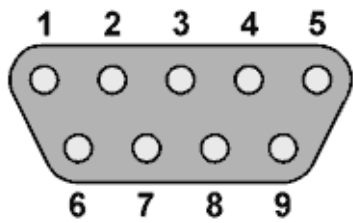


Интерфейс RS-232, более известный как интерфейс "COM-порта", ранее был одним из самых распространенных интерфейсов в компьютерной технике. Он до сих пор встречается в настольных компьютерах, несмотря на появление более скоростных и "интеллектуальных" интерфейсов, таких как USB и FireWare.

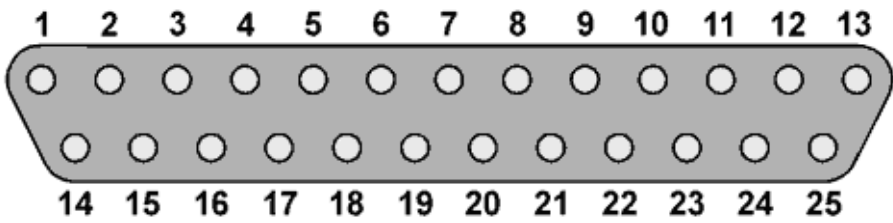
Физический интерфейс реализуется одним из двух типов разъемов: DB-9M или DB-25M, последний в выпускаемых в настоящее время компьютерах практически не встречается.



9-контактная вилка типа DB-9M  
Нумерация контактов со стороны штырьков  
Направление сигналов указано относительно хоста (компьютера)

Контакт	Сигнал	Направление	Описание
1	CD	Вход	Обнаружена несущая
2	RXD	Вход	Принимаемые данные
3	TXD	Выход	Передаваемые данные
4	DTR	Выход	Хост готов
5	GND	–	Общий провод
6	DSR	Вход	Устройство готово
7	RTS	Выход	Хост готов к передаче
8	CTS	Вход	Устройство готово к приему
9	RI	Вход	Обнаружен вызов

Из таблиц видно, что 25-контактный интерфейс отличается наличием полноценного второго канала приема-передачи (сигналы, обозначенные "#2"), а также многочисленных дополнительных управляющих и контрольных сигналов. Однако, часто, несмотря на наличие в компьютере "широкого" разъема, дополнительные сигналы на нем просто не подключены.



25-контактная вилка типа DB-25M  
Нумерация контактов со стороны штырьков  
Направление сигналов указано относительно хоста (компьютера)

Контакт	Сигнал	Направление	Описание
1	SHIELD	–	Экран
2	TXD	Выход	Передаваемые данные
3	RXD	Вход	Принимаемые данные
4	RTS	Выход	Хост готов к передаче
5	CTS	Вход	Устройство готово к приему
6	DSR	Вход	Устройство готово
7	GND	–	Общий провод
8	CD	Вход	Обнаружена несущая
9	–	–	Резерв
10	–	–	Резерв
11	–	–	Не используется
12	SCD	Вход	Обнаружена несущая #2
13	SCTS	Вход	Устройство готово к приему #2
14	STXD	Выход	Передаваемые данные #2
15	TRC	Вход	Тактирование передатчика
16	SRXD	Вход	Принимаемые данные #2
17	RCC	Вход	Тактирование приемника
18	LLOOP	Выход	Локальная петля
19	SRTS	Выход	Хост готов к передаче #2
20	DTR	Выход	Хост готов
21	RLOOP	Выход	Внешняя петля
22	RI	Вход	Обнаружен вызов
23	DRD	Вход	Определена скорость данных
24	TRCO	Выход	Тактирование внешнего передатчика
25	TEST	Вход	Тестовый режим

**CD** – Устройство устанавливает этот сигнал, когда обнаруживает несущую в принимаемом сигнале. Обычно этот сигнал используется модемами, которые таким образом сообщают хосту о обнаружении работающего модема на другом конце линии.

**RXD** – Линия приема хостом данных от устройства.

**TXD** – Линия передачи хостом данных к устройству.

**DTR** – Хост устанавливает этот сигнал, когда готов к обмену данными. Фактически сигнал устанавливается при открытии порта коммуникационной программой и остается в этом состоянии все время, пока порт открыт.

**RTS** – Хост устанавливает этот сигнал перед тем, как начать передачу данных устройству, а также сигнализирует о готовности к приему данных от устройства. Используется при аппаратном управлении обменом данными.

**RI** – Устройство (обычно модем) устанавливает этот сигнал при получении вызова от удаленной системы, например при приеме телефонного звонка, если модем настроен на прием звонков.

**DSR** – Устройство устанавливает этот сигнал, когда включено и готово к обмену данными с хостом. Этот и предыдущий (DTR) сигналы должны быть установлены для обмена данными.

**CTS** – Устройство устанавливает этот сигнал в ответ на установку хостом предыдущего (RTS), когда готово принять данные (например, когда предыдущие присланные хостом данные переданы модемом в линию или есть свободное место в промежуточном буфере).

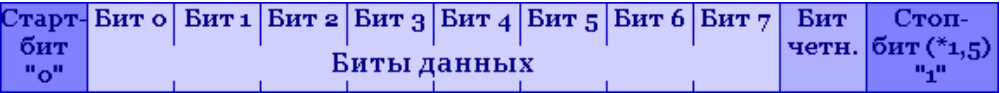
В протоколе RS-232 существуют два метода управления обменом данными: аппаратный и программный, а также два режима передачи: синхронный и асинхронный. Протокол позволяет использовать любой из методов управления совместно с любым режимом передачи. Также допускается работа без управления потоком, что подразумевает постоянную готовность хоста и устройства к приему данных, когда связь установлена (сигналы DTR и DSR установлены).

**Программный метод управления** заключается в передаче принимающей стороной специальных символов остановки (символ с кодом 0x13, называемый XOFF) и возобновления (символ с кодом 0x11, называемый XON) передачи. При получении данных символов передающая сторона должна соответственно остановить передачу или возобновить ее (при наличии данных, ожидающих передачи). Этот метод проще с точки зрения реализации аппаратуры, однако обеспечивает более медленную реакцию и соответственно требует заблаговременного извещения передатчика при уменьшении свободного места в приемном буфере до определенного предела.

**Аппаратный метод управления** реализуется с помощью сигналов RTS и CTS. Для передачи данных хост (компьютер) устанавливает сигнал RTS и ждет установки устройством сигнала CTS, после чего начинает передачу данных до тех пор, пока сигнал CTS установлен. Сигнал CTS проверяется хостом непосредственно перед началом передачи очередного байта, поэтому байт, который уже начал передаваться, будет передан полностью независимо от значения CTS. В полудуплексном режиме обмена данными (устройство и хост передают данные по очереди, в полнодуплексном режиме они могут делать это одновременно) снятие сигнала RTS хостом означает его переход в режим приема.

**Синхронный режим передачи** подразумевает непрерывный обмен данными, когда биты следуют один за другим без дополнительных пауз с заданной скоростью. Этот режим COM-портом **не поддерживается**.

**Асинхронный режим передачи** состоит в том, что каждый байт данных (и бит контроля четности, в случае его наличия) "оборачивается" синхронизирующей последовательностью из одного нулевого старт-бита и одного или нескольких единичных стоп-битов. Схема потока данных в асинхронном режиме представлена на рисунке.



**Один из возможных алгоритмов работы приемника** следующий:

1. Ожидать уровня "0" сигнала приема (RXD в случае хоста, TXD в случае устройства).
2. Отсчитать половину длительности бита и проверить, что уровень сигнала все еще "0"
3. Отсчитать полную длительность бита и текущий уровень сигнала записать в младший бит данных (бит 0)
4. Повторить предыдущий пункт для всех остальных битов данных
5. Отсчитать полную длительность бита и текущий уровень сигнала использовать для проверки правильности приема с помощью контроля четности
6. Отсчитать полную длительность бита и убедиться, что текущий уровень сигнала "1".
7. Вернуться к ожиданию начала следующего байта данных (шаг 1)

**Протокол имеет ряд переменных параметров**, которые должны быть приняты одинаковыми на стороне приемника и на стороне передатчика для успешного обмена данными:

- **Скорость обмена данными** задается в битах в секунду, определяя длительность одного бита, выбирается из ряда стандартных значений (300, 1200, 2400, 4800, 9600, 19200, 38400, 57600), но могут быть и нестандартными, если поддерживаются обеими сторонами;
- **Количество бит данных** может быть от 4 до 8;
- **Контроль четности** может быть четным ("even", когда общее число единичных битов в принятых данных, включая сам бит четности, должно быть четным), нечетным ("odd", когда общее число единичных битов в принятых данных, включая сам бит четности, должно быть четным) или вообще отсутствовать;
- **Длина стоп-бита** может составлять одну, полторы или две длительности бита.