

Методы передачи данных в компьютерных сетях

Существуют два способа передачи слов информации по линиям данных:

- параллельный, когда одновременно пересылаются все биты слова, и
- последовательный, когда биты пересылаются поочередно.

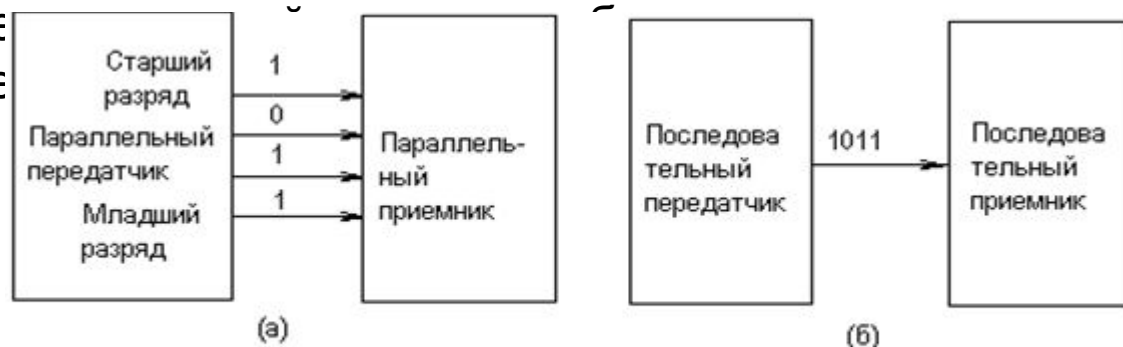


Рисунок 1 - Передача комбинации 1011_2 . (а) параллельная, (б) последовательная

Достоинства параллельной передачи данных.

Высокая скорость передачи данных.

Недостатки параллельной передачи данных.

Так как между отдельными проводниками шины для параллельной передачи данных существует электрическая емкость, то при изменении сигнала, передаваемого по одному из проводников, возникает помеха (короткий выброс напряжения) на других проводниках. С увеличением длины шины, а следовательно, с увеличением емкости проводников помехи возрастают и могут восприниматься приемником как сигналы. Поэтому рабочее расстояние для шины параллельной передачи данных ограничивается длиной 1-2 м.

Последовательная передача данных является преодолением ограничений, присущих параллельному способу передачи данных.

При последовательном обмене данными используются три метода передачи данных:

- симплексная (однаправленная) передача (телевидение, радио);
- полудуплексная (прием и передача информации осуществляются поочередно);
- дуплексная (двухнаправленная) каждая станция одновременно передает и принимает

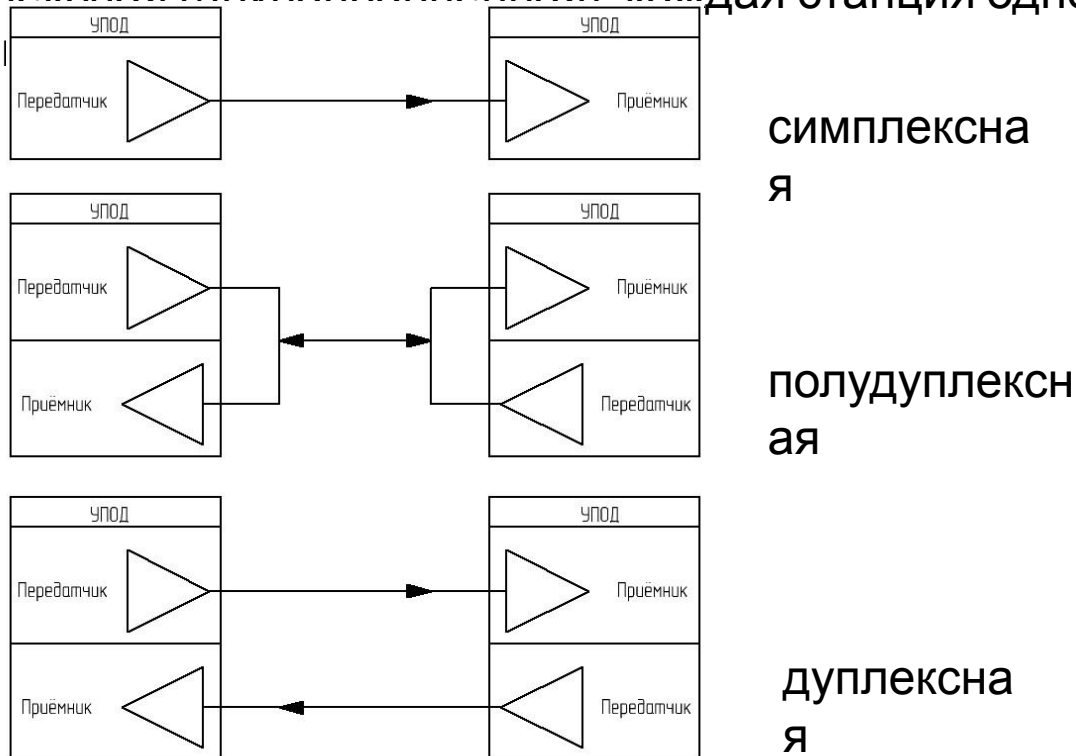


Рисунок 2 - Методы последовательной передачи данных

Прежде чем начать последовательную передачу данных, необходимо выполнить преобразование данных из параллельной формы в последовательную.



Рисунок 3 - Преобразование параллельных данных в последовательные

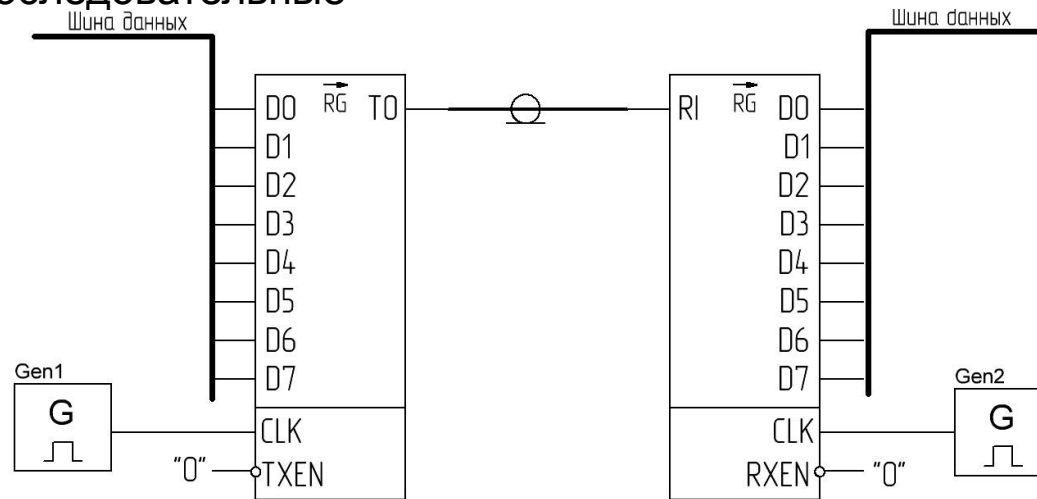


Рисунок 4 - Принцип последовательной передачи данных

Для передачи данных в сетях наиболее часто применяется последовательная передача. Широко используются следующие методы последовательной передачи:

- асинхронная и
- синхронная

Последовательная асинхронная передача данных

Суть асинхронного принципа состоит в независимой (полностью или частично) работе (по времени) передатчика и приемника, управляемой регулярной последовательностью сигналов синхронизации при реализации функций обмена и обработки информации.

Наибольшее применение получили стартстопные принципы синхронизации по битам и знакам.

Суть стартстопного принципа управления состоит в том, что стартовый импульс в сообщении запускает местный синхрогенератор приемника, который работает на частоте передатчика, и линия стробируется в соответствии с частотой местного синхронизатора, а стоповый импульс в сообщении останавливает синхрогенератор.

Передача данных осуществляется порциями (кадрами). Начало и конец каждой порции информации отмечаются специальными метками.

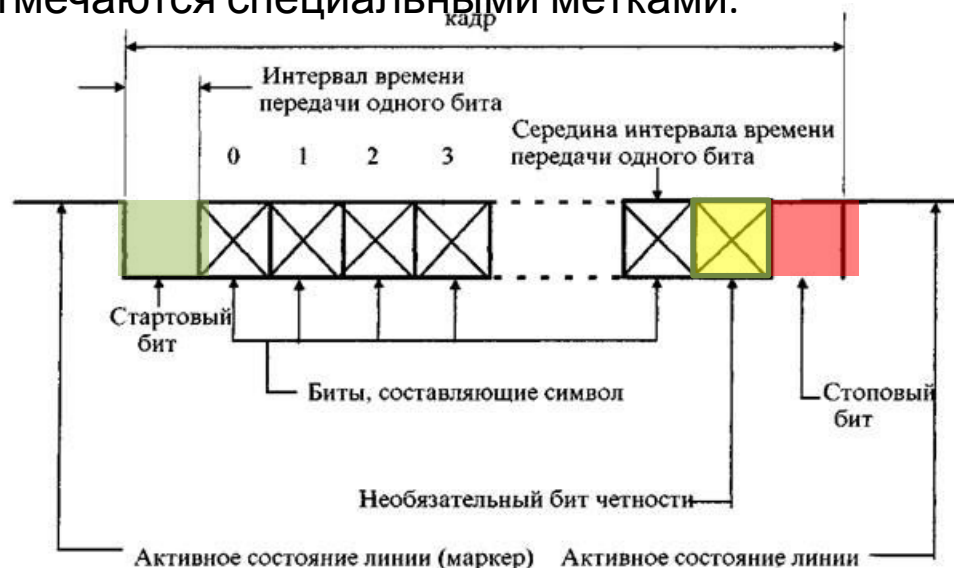


Рисунок 8 - Формат кадра при асинхронной последовательной передаче

- Уровень логической единицы в линии называют маркером,
- уровень логического нуля - пробелом.
 - При отсутствии данных в линии действует сигнал маркера.
 - Передача кадра начинается с посылки стартового бита - пробела.
Стартовый бит предупреждает приемник о начале передачи.
 - После стартового бита передаются биты данных, число которых в кадре может устанавливаться от 5 до 8.
 - Для определения достоверности передачи данных после последнего бита данных может следовать бит паритета, который также называется битом контроля четности (или нечетности). Этот бит выбирается в каждом кадре данных таким образом, чтобы общее число единиц в битах данных и бите паритета было четным (или нечетным), т.е. бит четности = 1, если количество единиц в символе нечетно, и 0 в противном случае
 - Кадр заканчивается стоп-битами, которые сигнализируют об окончании передачи и имеют уровень маркера.
Может устанавливаться один, полтора или два стоповых бита.
 - После этого в линии может поддерживаться состояние отсутствия данных (уровень маркера) либо начинаться следующий кадр (стартовым битом - пробелом).

Асинхронный приёмопередатчик

Устройство, обеспечивающее преобразование данных из параллельной формы в последовательную и обратно и обеспечивает асинхронный протокол обмена данными, называют универсальным асинхронным приёмопередатчиком (УАПП, UART).

Такое устройство реализуется в виде специального модуля микроконтроллера или отдельной БИС. Кроме преобразования формы представления приёмопередатчик выполняет важные функции контроля и управления.

Скорость передачи данных описанным способом принято измерять в бодах.

Один бод равен одному биту в секунду, и поэтому скорость в бодах выражает, сколько бит в секунду может быть передано по соответствующему каналу передачи данных.

Организация асинхронного последовательного обмена данными с внешним устройством осложняется тем, что на передающей и приемной стороне последовательной линии связи используются настроенные на одну частоту, но физически разные генераторы тактовых импульсов и, следовательно, общая синхронизация отсутствует.

Для решения этой проблемы для работы УАПП используется последовательность тактовых импульсов, период следования которых в 16 раз меньше времени передачи одного бита данных.

Эти тактовые импульсы используются для синхронизации входного сигнала и для

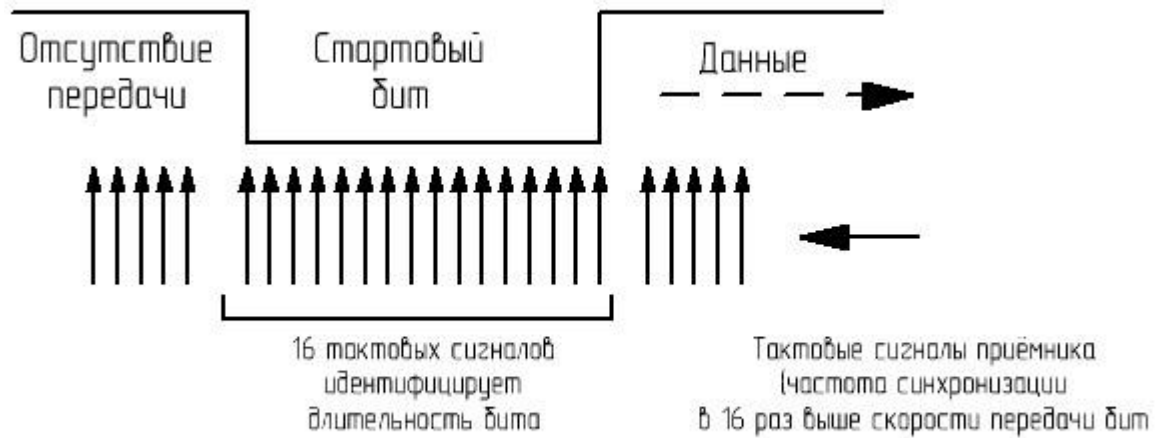


Рисунок 9 - Извлечение синхросигнала.

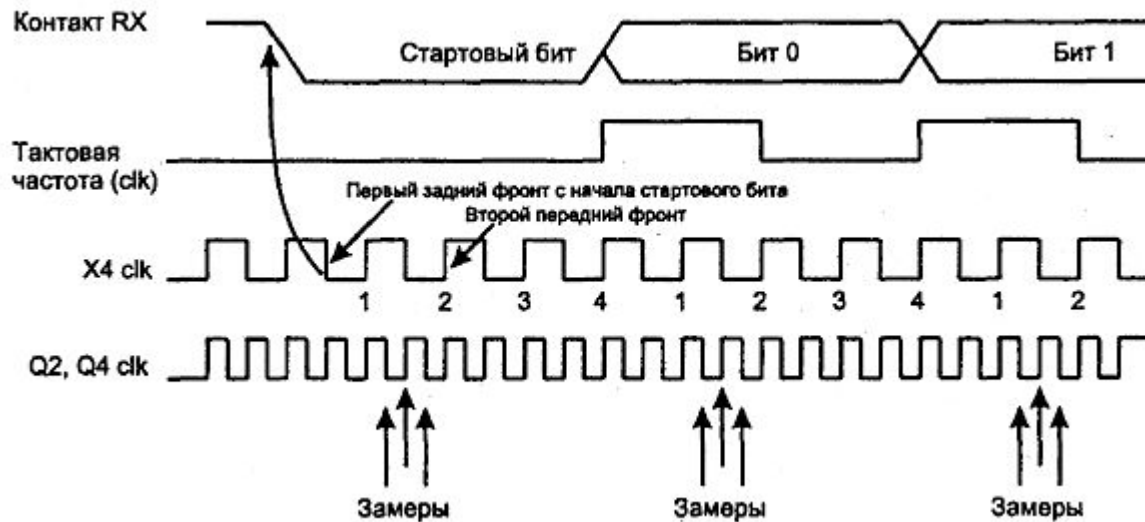


Рисунок 10 - Временная диаграмма работы приемника UART

В схеме можно выделить 4 основные части:

- приёмную,
- передающую,
- регистр состояния и
- блок управления.

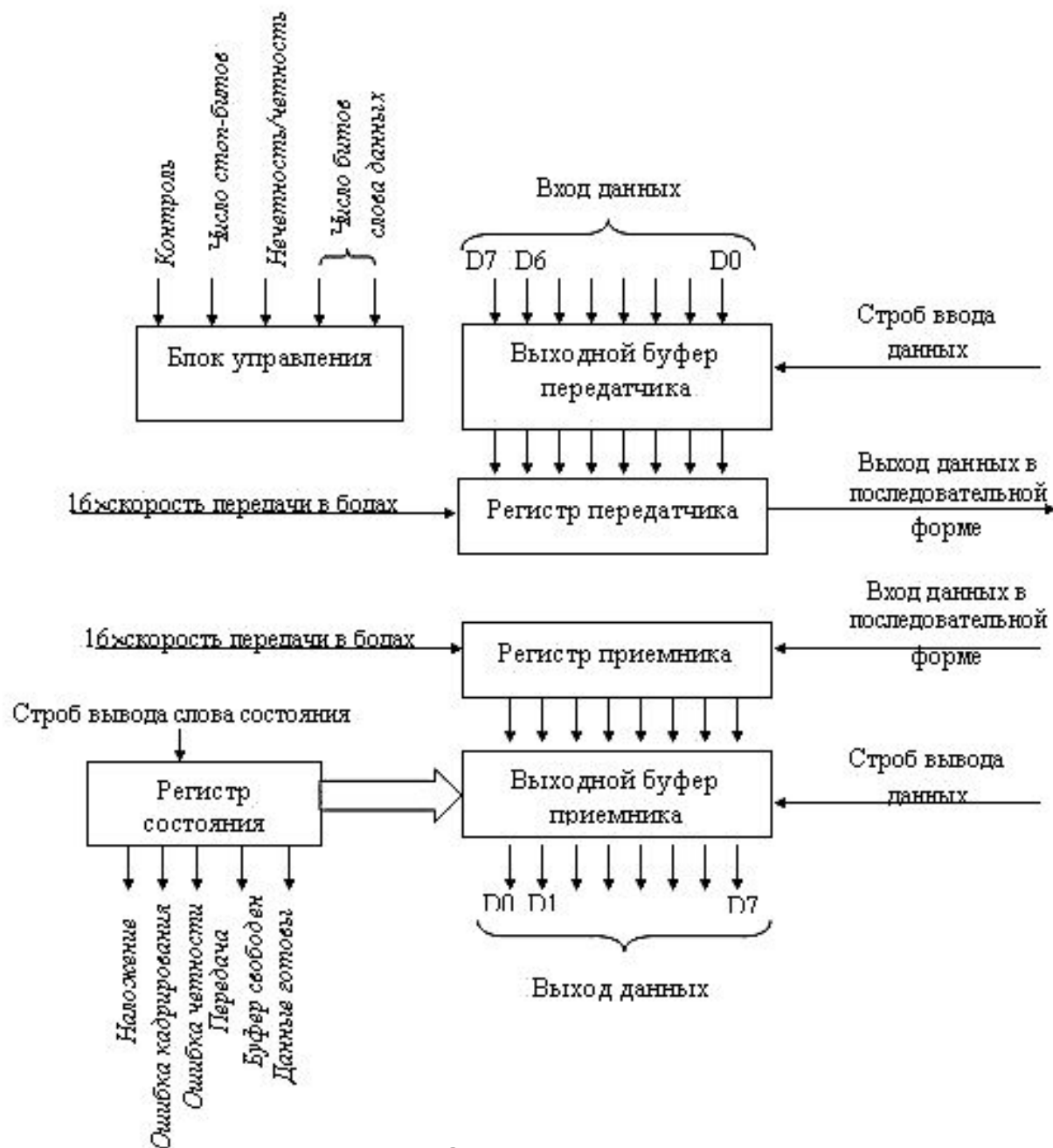


Рисунок 11 - Структурная схема простого УАПП

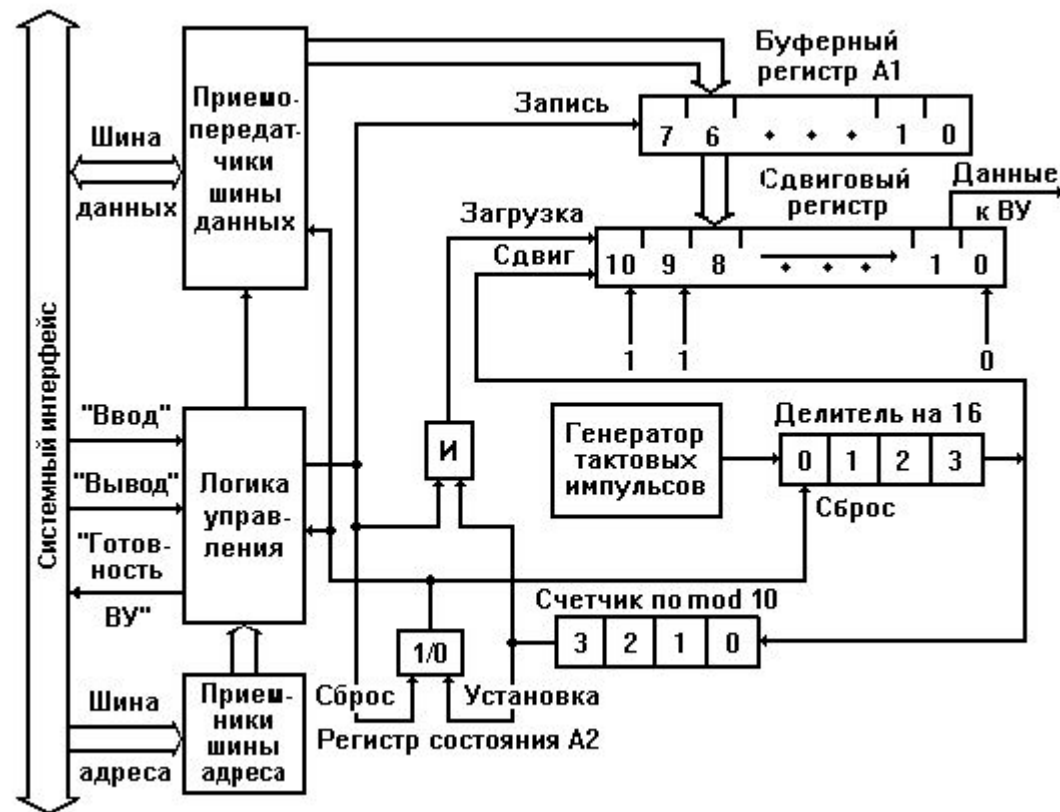


Рисунок 12 - Контроллер последовательной асинхронной передачи

Процесс передачи байта данных начинается с того, что процессор, выполняя команду "Вывод", выставляет этот байт на шине данных. Одновременно процессор формирует управляющий сигнал системного интерфейса "Вывод", по которому производятся запись передаваемого байта в буферный регистр A1, сброс регистра состояния A2 и формирование на вентиле И сигнала "Загрузка".

Передаваемый байт переписывается в разряды 1, ... , 8 сдвигового регистра, в нулевой разряд сдвигового регистра записывается 0 (стартовый бит), а в разряды 9 и 10 - 1 (стоповые биты). Кроме того, снимается сигнал "Сброс" с делителя частоты на 16, он начинает накапливать импульсы генератора тактовой частоты и в момент приема шестнадцатого тактового импульса вырабатывает импульс сдвига.

На выходной линии контроллера "Данные" поддерживается состояние 0 (значение стартового бита) до тех пор, пока не будет выработан первый импульс сдвига.

Импульс сдвига изменит состояние счетчика импульсов сдвига и перепишет в нулевой разряд сдвигового регистра первый информационный бит передаваемого байта данных. Состояние, соответствующее значению этого бита, будет поддерживаться на линии "Данные" до следующего импульса сдвига.

Аналогично будут переданы остальные информационные биты, первый стоповый бит и, наконец, второй стоповый бит, при передаче которого счетчик импульсов сдвига снова установится в нулевое состояние. Это приведет к записи 1 в регистр состояния A2.

Единичный сигнал с выхода регистра A2 запретит формирование импульсов сдвига, а также информирует процессор о готовности к приему нового байта данных.

После завершения передачи очередного кадра (стартового бита, информационного байта и двух стоповых бит) контроллер поддерживает в линии связи уровень логической единицы (значение второго стопового бита).

После передачи очередного байта данных в регистр состояния A2 записывается 1, который информирует процессор о готовности контроллера к приему следующего байта данных и передаче его по линии связи в ВУ. Этот же сигнал запрещает формирование импульсов со схемы выработки импульсов сдвига - делителя частоты сигналов тактового генератора на 16. Счетчик импульсов сдвига (счетчик по mod 10) находится в нулевом состоянии и его единичный выходной сигнал поступает на вентиль И, подготавливая цепь выработки сигнала загрузки сдвигового регистра.

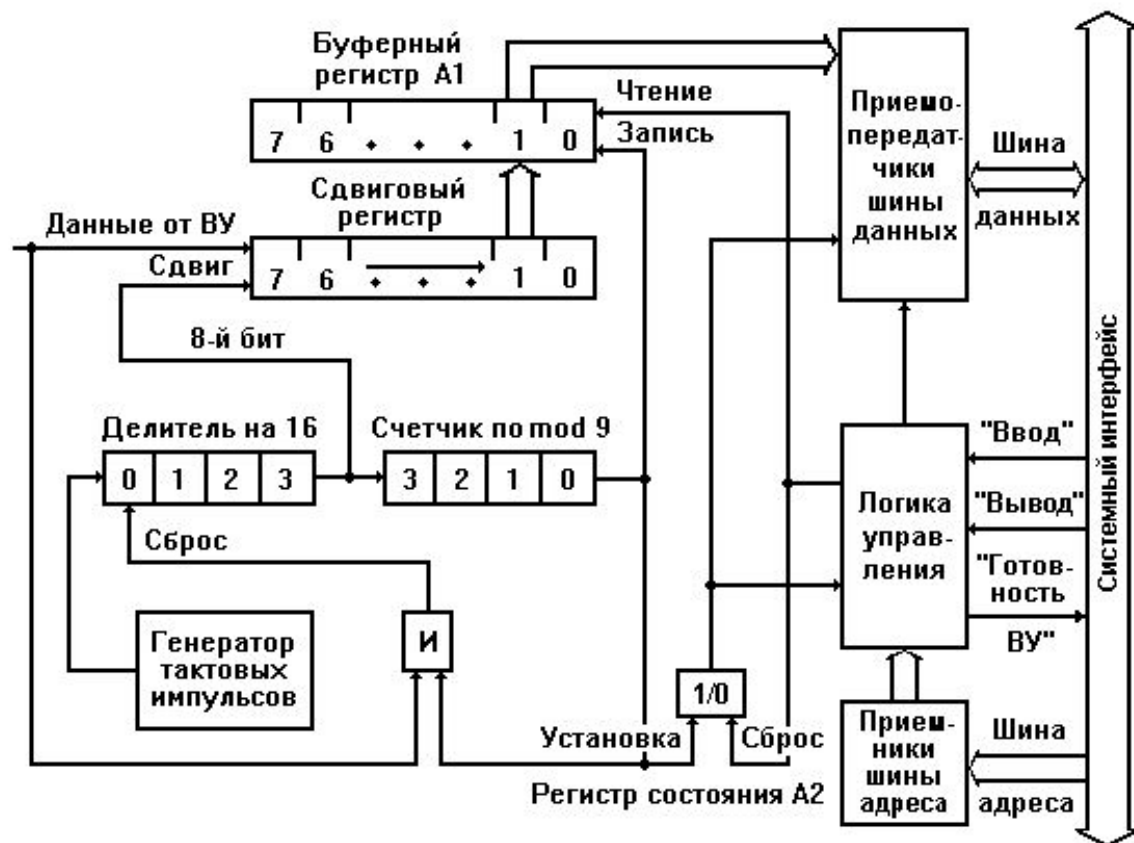


Рисунок 13 - Контроллер последовательного асинхронного приема

Уровень логической единицы поступает по линии "Данные" в контроллер для асинхронного приема данных (рис. 13). Этот уровень создает условия для выработки сигнала, запрещающего работу делителя частоты генератора тактовых импульсов.

Действительно, после приема предыдущего байта данных счетчик импульсов сдвига (счетчик по mod 9) находится в нулевом состоянии и на вентиль И поступают два единичных сигнала: со счетчика сдвигов и из линии "Данные". На выходе вентиля И вырабатывается сигнал сброса делителя частоты сигналов тактового генератора, запрещающий формирование импульсов сдвига.

В момент смены стопового бита на стартовый бит (момент начала передачи нового кадра) на линии "Данные" появится уровень логического нуля и тем самым будет снят сигнал сброса с делителя частоты.

Состояние 4-разрядного двоичного счетчика (делителя частоты) начнет изменяться. Когда на счетчике накопится значение 8, он выдаст сигнал, поступающий на входы сдвигового регистра и счетчика импульсов сдвига. Так как частота сигналов генератора тактовых импульсов приемника должна совпадать с частотой генератора тактовых импульсов передатчика, то сдвиг (считывание) бита произойдет примерно на середине временного интервала, отведенного на передачу бита данных, т. е. времени, необходимого для выработки шестнадцати тактовых импульсов. Это делается для уменьшения вероятности ошибки из-за возможного различия частот генераторов передатчика и приемника, искажения формы передаваемых сигналов (переходные процессы) и т. п. Следующий сдвиг произойдет после прохождения шестнадцати тактовых импульсов, т. е. на середине временного интервала передачи первого информационного бита.

При приеме в сдвиговый регистр девятого бита кадра (восьмого информационного бита) из него "выдвинется" стартовый бит и, следовательно, в сдвиговом регистре будет размещен весь принятый байт информации. В этот момент счетчик импульсов сдвига придет в нулевое состояние и на его выходе будет выработан единичный сигнал, по которому содержимое сдвигового регистра переписывается в буферный регистр, в регистр состояния A2 запишется 1 и он будет информировать процессор об окончании приема очередного байта, вентиль И подготовится к выработке сигнала "Сброс" (этот сигнал сформируется после прихода первого стопового бита).

Получив сигнал готовности (1 в регистре A2), процессор выполнит команду "Ввод" (см. пример 2.2 Параллельной передачи данных). При этом вырабатывается управляющий сигнал системного интерфейса "Ввод", по которому производится пересылка принятого байта данных из буферного регистра в процессор (сигнал "Чтение") и сброс регистра состояния A2

Преимущества:

- несложная отработанная система;
- недорогое интерфейсное оборудование.

Недостатки асинхронной передачи:

- третья часть пропускной способности теряется на передачу служебных битов (старт/стоповых и бита четности);
- Относительно невысокая скорость передачи (по сравнению с синхронной);
- при множественной симметричной ошибке в битах данных с помощью бита четности невозможно определить достоверность полученной информации.

Асинхронная передача используется в системах, где обмен данными происходит время от времени и не требуется высокая скорость передачи данных.

Некоторые системы используют бит четности как символьный бит, а контроль информации выполняется на уровне протоколов обмена данными.

Рекомендованный стандарт RS232

Интерфейс между терминалом
данных и передающим
оборудованием линии связи,
применяющий последовательный
обмен двоичными данными.

RS-232 (Recommended Standard 232) - стандарт описывающий интерфейс для последовательной двунаправленной передачи данных между терминалом (DTE, Data Terminal Equipment) и конечным устройством (DCE, Data Circuit-Terminating Equipment).

Появился в 60-х годах XX века, и стал основой для всех последующих интерфейсов последовательного обмена данными.

Интерфейс RS-232C был применен в первых персональных компьютерах фирмы IBM и до сегодняшнего дня входит в структуру любого персонального компьютера в аппаратном или программном виде.

Решения, которые заложены в этот стандарт, используются практически повсеместно.

Интерфейс RS-232 полностью аппаратно реализован на персональных компьютерах в виде микросхем и разъемов. Аппаратная реализация означает то, что он работает всегда, не зависимо, какая операционная система установлена на PC (он работает и без ОС). В PC его называют COM-портом (Communication port).

Программы могут взаимодействовать с COM-портами всеми доступными средствами: прямым кодом микропроцессора, аппаратными прерываниями, функциями BIOS, средствами ОС, компонентами языков высокого уровня.

COM порт реализованный по стандарту RS-232- универсален.

Он обеспечивает работу PC с:

- периферийными устройствами (чем сейчас занят USB),
- взаимодействие с локальной сетью через модем (Ethernet),
- обмен данными между PC и промышленным оборудованием (ModBus и др.).

Чтобы разбираться как работают эти протоколы необходимо понимать какую функцию COM порта они взяли на себя.

Содержание стандарта

RS-232

Оригинальная часть стандарта RS-232 содержит шесть секций.

1. Область применения.
2. Характеристики сигнала.
3. Механические характеристики интерфейса.
4. Функциональное описание сигналов обмена.
5. Стандартные интерфейсы для выбранных конфигураций системы связи.
6. Рекомендации и приложения.

Область применения RS-232

Стандарт RS-232 определяет взаимосвязи терминального оборудования ввода данных (DTE) и оконечного оборудования линии связи (DCE), использующих последовательный обмен двоичных данных.

Стандарт RS-232 включает тринадцать определенных конфигураций интерфейса, обозначенных латинскими буквами A-M и одной пользовательской конфигурацией, обозначаемой Z. Конфигурация интерфейса должна указываться производителем коммуникационного оборудования.

Конфигурация интерфейса определяет состав сигналов, синхронность, порядок выставления сигналов, скорость и др.

Стандарт RS-232 применяется в системах с общими электрическими проводами и не может применяться в системах где требуется гальваническая развязка цепей.

Стандарт RS-232 может применяться в синхронных и асинхронных системах последовательной передачи двоичных данных.

Характеристики сигналов

RS-232C

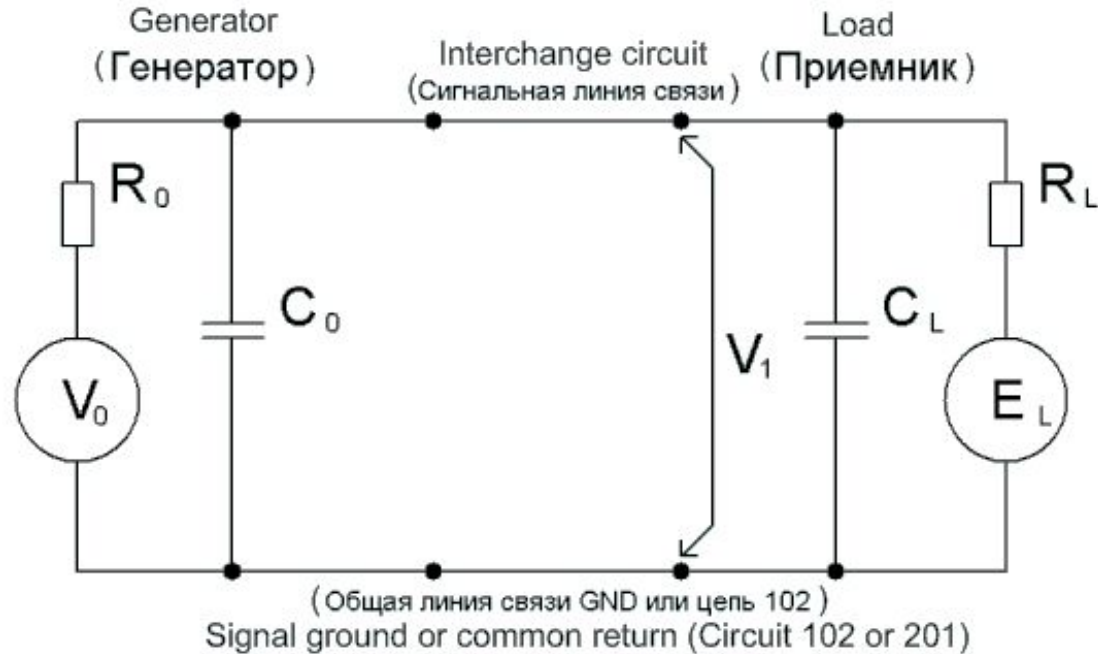


Рис.1 Эквивалентная электрическая схема
RS-232C

V_0 - напряжение генератора при разомкнутой схеме

R_0 - общее сопротивление генератора

C_0 - общая ёмкость генератора

V_1 - напряжение между сигнальной линией и
общим проводом в месте стыка.

C_L - общая ёмкость приёмника

R_L - общее сопротивление приёмника

E_L - ЭДС приёмника при разомкнутой схеме

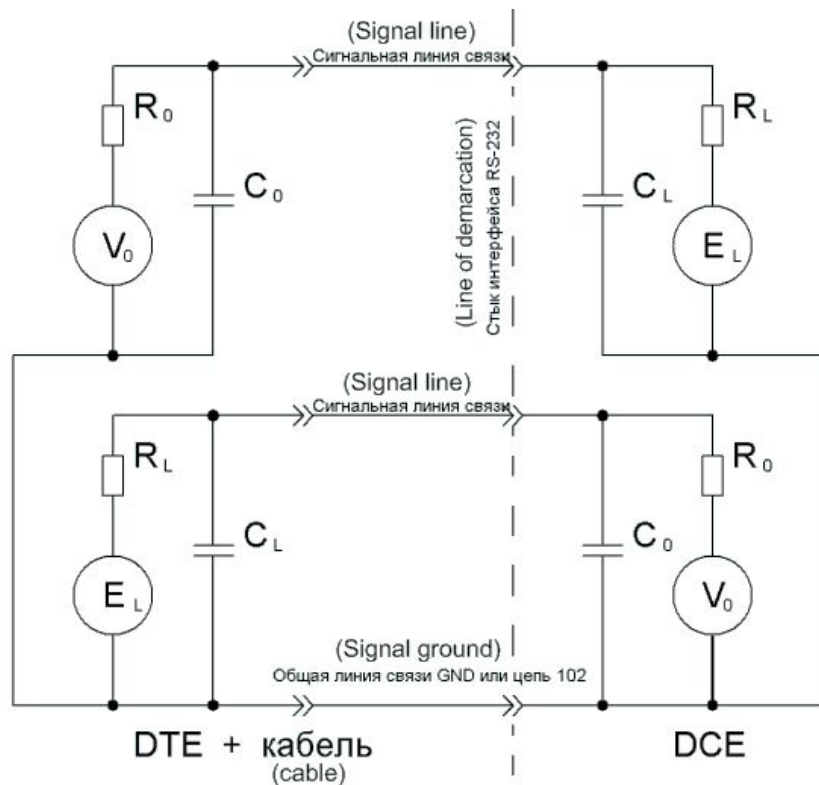
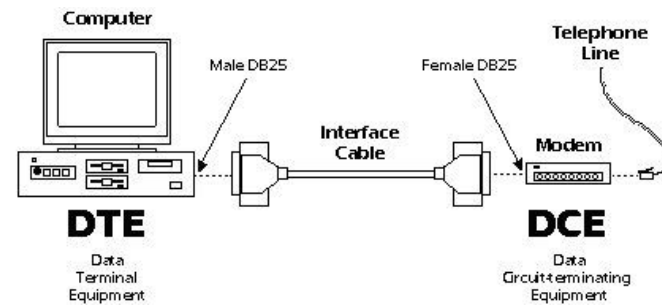


Рис.2 Практическая схема стыка интерфейса RS-232C

Электрические характеристики генератора сигналов

- Допускается короткое замыкание сигналов.
- Допускается оставлять выход генератора без нагрузки.
- V_0 - напряжение генератора при разомкнутой схеме должно быть не более **$\pm 25\text{В}/\pm 15\text{В}$** (RS-232/ITU-T v.28)
- R_0 и C_0 для генератора не нормируются.
- Короткое замыкание цепей генератора не должно вызывать токи величиной более 0,5А.
- Если $E_L = 0$, то напряжение на входе приёмника должно быть $V_1 = \pm 5... \pm 15\text{В}$, для любого диапазона нагрузки генератора $R_L = 3000...7000\text{Ом}$.
- Генератор должен быть способен работать на ёмкостную нагрузку C_0 плюс **2500 пФ**.



Электрические характеристики приёмника сигналов

- R_L - общее сопротивление приёмника должно находиться в пределах 3000...7000 Ом.
- V_1 - напряжение на входе приёмника должно быть в пределах $\pm 3... \pm 15\text{В}$.
- E_L - ЭДС приёмника при разомкнутой схеме должно быть не более $\pm 2\text{В}$.
- C_L - общая ёмкость цепей приёмника должна быть не более 2500 пФ.
- Входной импеданс приёмника не должен быть индуктивным.

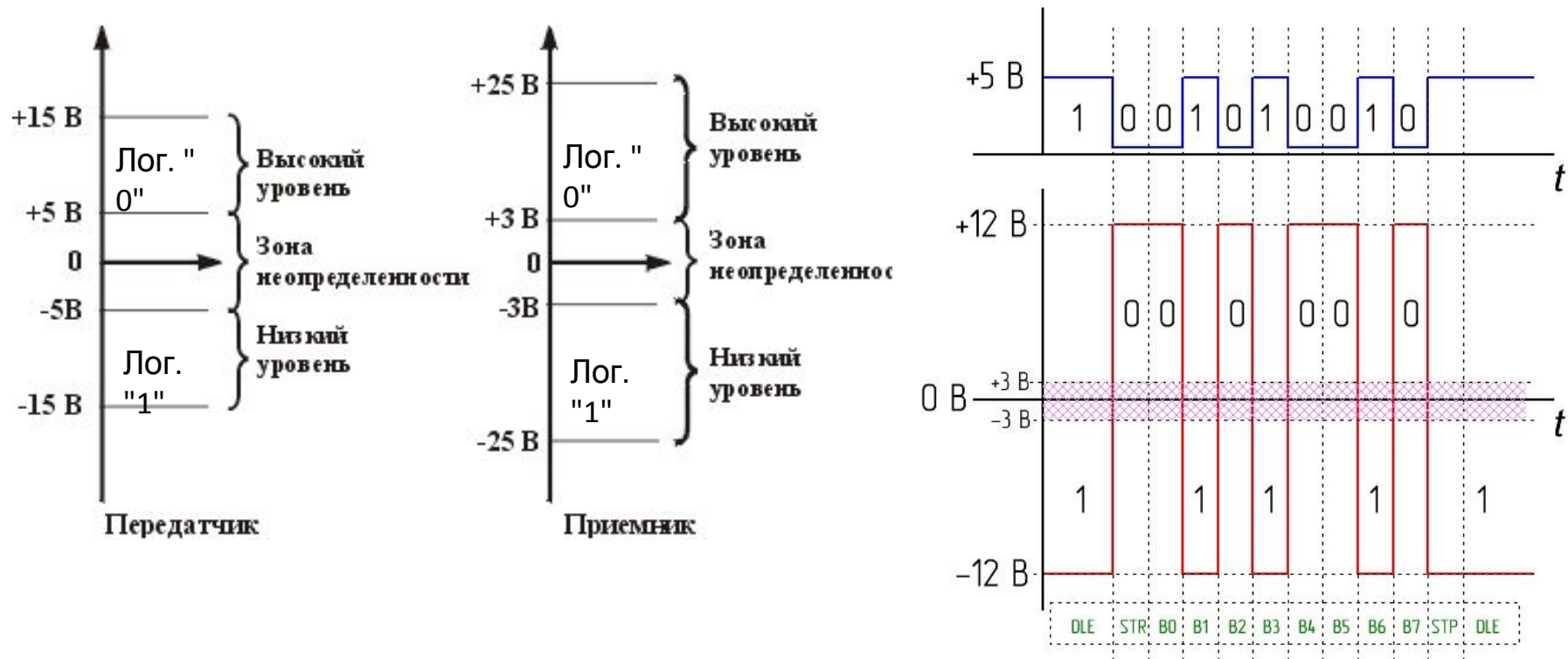


Рис.3 Уровни сигналов RS-232C на передающем и принимающем концах линии связи.

Уровни сигналов для стандарта RS-232C

Логической "1" считается информационный сигнал с напряжением V_1 менее -3 В.

Логическим "0" считается информационный сигнал с напряжением V_1 более +3 В.

Напряжение в диапазоне $V_1 = -3 \text{ В} \dots +3 \text{ В}$ считается переходной областью

Механические характеристики интерфейса.

Разъем для интерфейса стандарта RS-232

В качестве разъема для интерфейса RS-232C выбран миниатюрный разъем D-типа (D-subminiature).

Обозначение разъемов D-subminiature:

Буква - тип разъема. D-тип

Буква - типоразмер обоймы. A, B, C, D, E

Число - количество контактов в обойме

Буква - тип контакта. P = plug(штырь), S = socket(гнездо).

Для терминалов (DTE) - **DB25p**

Для оконечных устройств (DCE) - **DB25s**

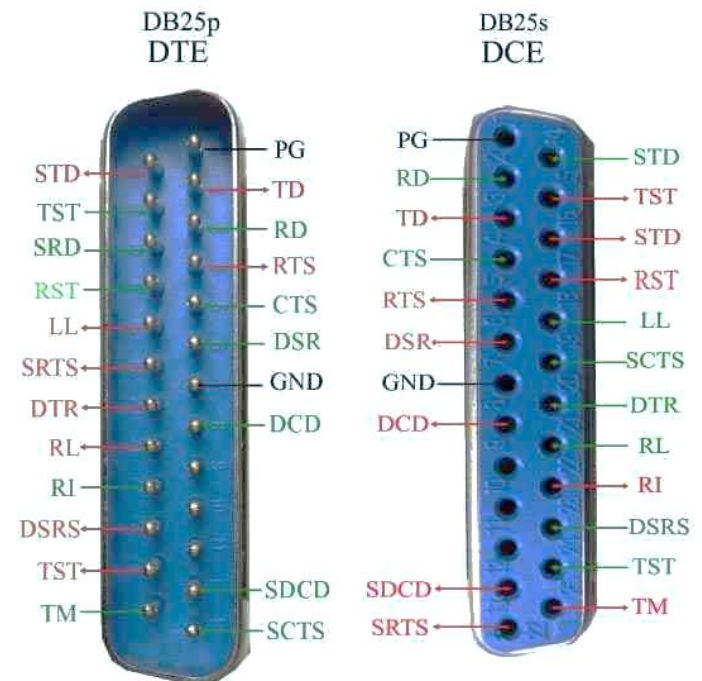
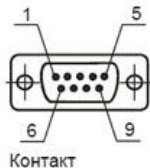
При использовании стандарта TIA/EIA 574 можно применять 9-ти штырьковые разъемы:

Для терминалов (DTE) - **DE9p**

Для оконечных устройств (DCE) - **DE9s**

Com порт - rs232

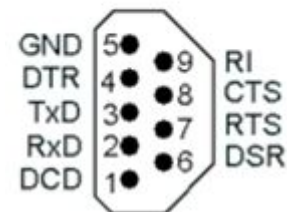
Разъем 9-Pin M



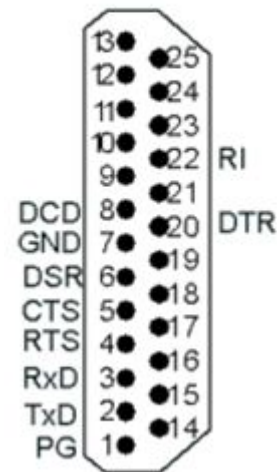
Назначение сигналов COM порта по стандарту RS-232C

Наименование	Направление	Описание	DB25	DB9	Назначение
DCD	IN	Carrie Detect (Определение <u>несушей</u>)	8	1	детектирование принимаемого сигнала
RXD	IN	Receive Data (Принимаемые данные)	3	2	данные, принимаемые компьютером в последовательном коде (логика отрицательная)
TXD	OUT	Transmit Data (Передаваемые данные)	2	3	данные, передаваемые компьютером в последовательном коде (логика отрицательная).
DTR	OUT	Data Terminal Ready (Готовность терминала)	20	4	готовность выходных данных
GND	-	System Ground (Корпус системы)	7	5	сигнальное заземление, нулевой провод
DSR	IN	Data Set Ready (Готовность данных)	6	6	готовность данных. Используется для задания режима модема
RTS	OUT	Request to Send (Запрос на отправку)	4	7	сигнал запроса передачи. <u>Активен</u> во все время передачи.
CTS	IN	Clear to Send (Готовность приема)	5	8	сигнал сброса (очистки) для передачи. <u>Активен</u> во все время передачи. Говорит о готовности приемника
RI	IN	Ring Indicator (Индикатор вызова)	22	9	Говорит о приеме модемом сигнала вызова по телефонной сети.
FG	-	<u>защитное заземление.</u>			<u>экран</u>

DE9p



DB25p



Организация обмена данных при аппаратном режиме синхронизации.

Аппаратный режим синхронизации обмена данными RTS/CTS (hardware flow control)

Handshaking =2, использует сервисные сигналы RS-232C для управления потоком данных.

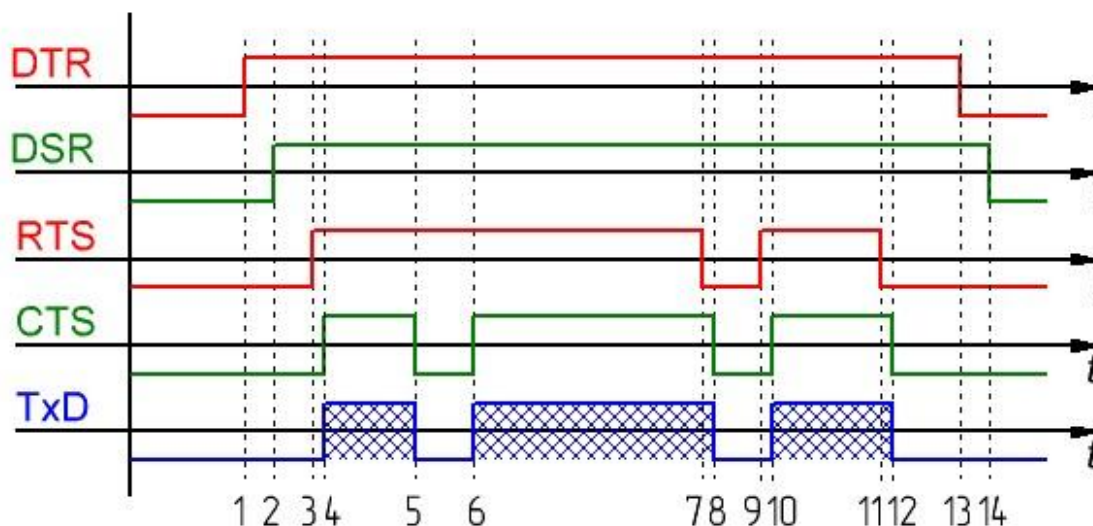


Рис.7 Организация обмена при аппаратной синхронизации.

1. DTR=True компьютер указывает на желание использовать модем.
2. В ответ модем сигнализирует о установлении соединения с другим модемом и своей готовности выставив DSR=True.
3. Сигналом RTS=True компьютер запрашивает разрешение на передачу и заявляет о своей готовности принимать данные от модема.
4. Сигналом CTS=True модем уведомляет о своей готовности к приему данных от компьютера и передаче их в линию. По этому сигналу начинается обмен данными между терминалами через модемы.
5. Установкой CTS=False модем сигнализирует о невозможности дальнейшего приема, компьютер должен приостановить передачу данных.
6. Установкой CTS=True, модем сообщает, что он может продолжать обмен данными.
7. Установкой RTS=False компьютер сообщает модему о временной приостановке обмена.
8. Модем получив сигнал остановки обмена RTS=False, сообщает о приостановки обмена сигналом CTS=False
9. Компьютер вновь готов принимать данные и он выставляет сигнал RTS=True
10. Модем получив сигнал от компьютера о готовности к обмену выставляет свою готовность CTS=True. После чего возобновляется обмен данными.
11. Компьютер указывает на завершение обмена выставив RTS=False
12. Модем подтверждает завершение обмена сигналом CTS=False
13. Компьютер снимает сигнал DTR, что является сообщением для модема разорвать соединение и повесить трубку.
14. Модем подтверждает разрыв соединения установкой сигнала DSR=False

Соединительные кабели

Нуль модемные кабели

RS-232

3-проводный минимальный

При таком соединении компьютеры (терминалы) соединяются между собой непосредственно через COM-порты, без использования модемов.

Так как компьютеры обладают большой скоростью обработки данных, то синхронизировать их работу не нужно.

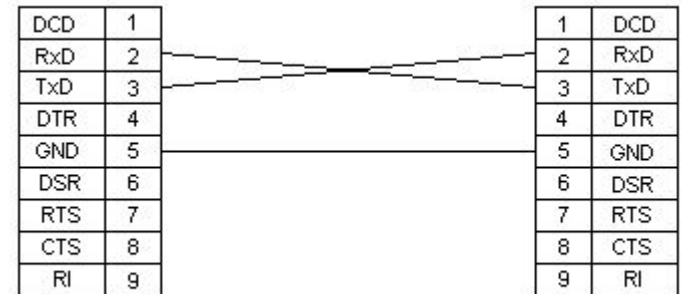
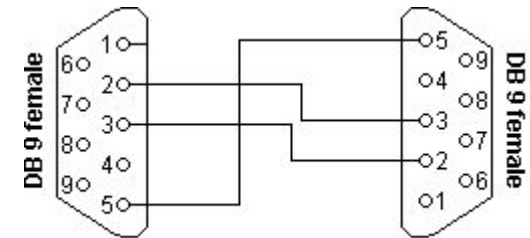


Рис.4 Нуль-модемный кабель для Handshaking = 0 (None)

Поэтому предполагается, что если режим синхронизации обмена (Handshaking):=0-None, то есть сервисные сигналы не влияют на процедуры обмена данными.

Для этого используется нуль-модемный кабель.

Так как режим синхронизации обмена на COM портах может быть включен, то часто сервисные сигналы COM портов замыкают самих на себя, тем самым исключая их влияния на процедуру обмена.

DSR сигнал (конт.6) это вход сигнала готовности от аппаратуры передачи данных. В схеме соединений вход замкнут на выход DTR (конт.4) (готовность выходных данных). Это означает, что программа не видит сигнала готовности другого устройства, хотя он есть.

Аналогично устанавливается сигнал на входе DCD (конт.1). Тогда при проверке сигнала DSR для контроля возможности соединения будет установлен выходной сигнал DTR.

Это соответствует 99% коммуникационного программного обеспечения. Под этим подразумевается, что 99% программного обеспечения с этим нуль-модемным кабелем примут проверку сигнала DSR.

Аналогичный для входного сигнала CTS. В оригинале сигнал RTS (конт.7) устанавливается и затем проверяется CTS (конт.8).

Соединение этих контактов приводит к

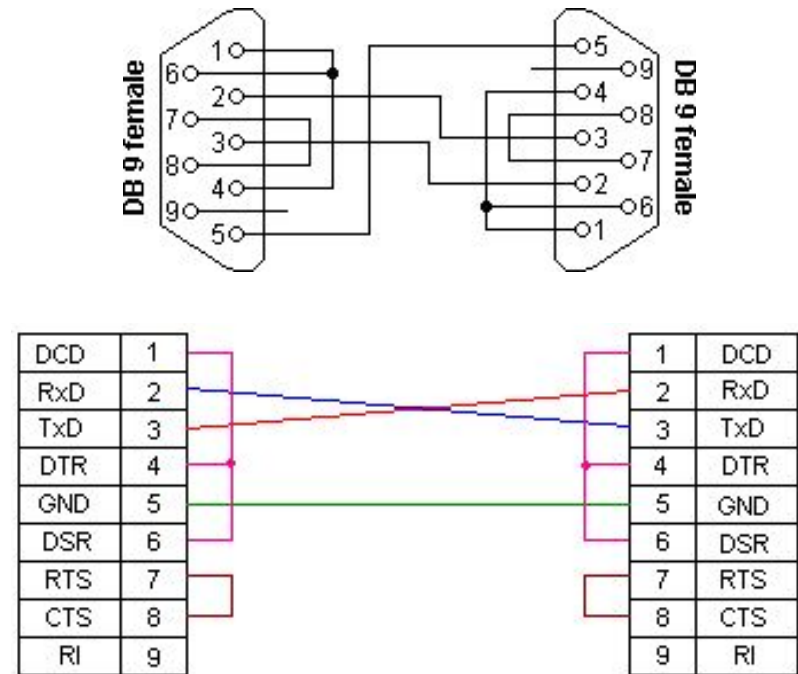


Рис.5 Нуль-модемный кабель для любых режимов Handshaking

7-проводный полный .

Самый дорогой полный нуль-модемный кабель с семью проводами.

Только сигналы индикатора вызова и определения несущей не подключены.

Совместимость.

Этот кабель не разрешает использовать предыдущий метод контроля передачи данных. Основная несовместимость перекрестное соединение сигналов RTS и CTS. Первоначально эти сигналы использовались для контроля потоком данных по типу запрос/ответ. При использовании полного нуль-модемного кабеля более нет запросов. Эти сигналы применяются для сообщения другой стороне есть ли возможность соединения.

Особенность.

Контакты 2 и 3 на 9-ти выводном разъеме D типа противоположны этим же контактам на 25-ти контактном разъеме. Поэтому, если соединить контакты 2-2 и 3-3 между разъемами D25 и D9, получится коммуникационный кабель. Контакты сигнальной земли Signal Ground (SG) также должны быть подключены между собой.

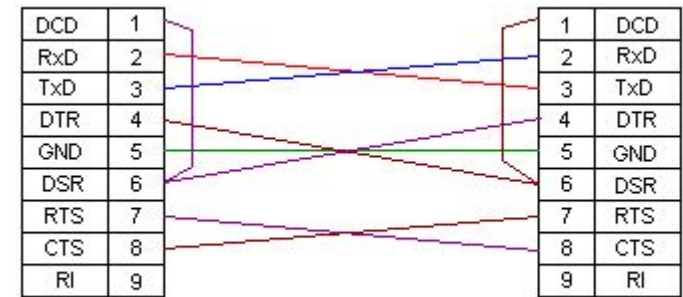
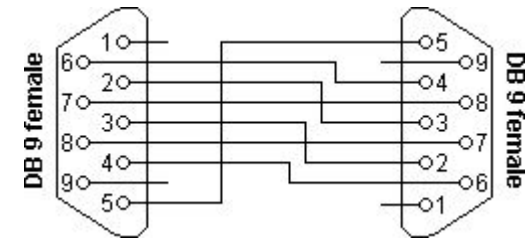


Рис.6 Нуль-модемный кабель для аппаратного режима Handshaking2

Заглушка тестирования RS-232

Заглушка для эмуляции терминала.

Данный соединитель RS-232 может быть использован для проверки последовательного порта компьютера. Сигналы данных и управления соединены. В этом случае передаваемые данные сразу возвращаются. Компьютер проверяет собственный поток. Это может быть использовано для проверки функционирования порта RS-232 со стандартным терминальным программным обеспечением.

Кабель контроля (мониторинга)

RS-232

Контроль связи по RS-232 между двумя устройствами с помощью компьютера возможен при помощи кабеля, изображенного на рисунке 7.

Два разъема подключаются к устройствам, а третий подключается к наблюдающему компьютеру. Этот кабель принимает информацию от двух источников только на один приемный порт RS-232. Поэтому, если оба устройства начнут одновременную работу, контролируемая информация на входе компьютера будет нарушена.

В большинстве случаев связь осуществляется в полудуплексном режиме. Для этих режимов этот кабель будет работать без проблем.

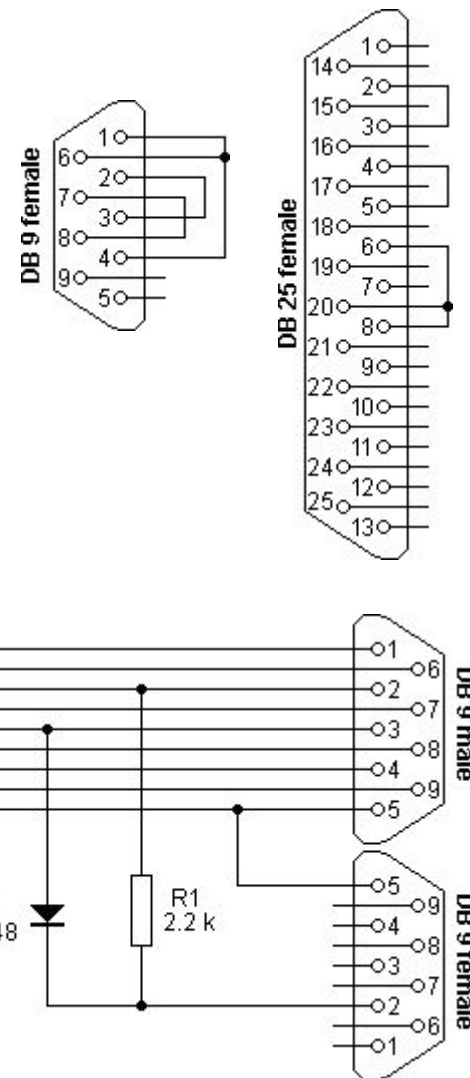


Рис.7 Кабель для наблюдения передачи данных в полудуплексном режиме

Расстояния

передачи

Длина кабеля. Длина кабеля влияет на максимальную скорость передачи информации. Более длинный кабель имеет большую емкость и соответственно для обеспечения надежной передачи более низкую скорость. Большая емкость приводит к тому, что изменение напряжения одного сигнального провода может передаться на другой смежный сигнальный провод. Максимальным расстоянием обычно считается равным 15 м, но это не установлено в стандарте. Рекомендуется использовать на расстояниях до 50 м, но это зависит от типа используемого оборудования и характеристик кабеля.

Максимальная длина кабеля

Скорость [бод]	Макс. длина [футы]	Макс. длина [метры]
19 200	50	15
9 600	500	150
4 800	1000	300
2 400	3000	900

Скорость передачи данных

Скорость передачи информации по RS-232 измеряется в Бодах. Эта единица названа в честь Эмиля Бодо (Jean Maurice-Emile Baudot) (1845-1903), французского инженера по телеграфии, изобретателя первого печатающего устройства для телеграфа (телепринтера), представленного на Международной Телеграфной конференции в 1927. Максимальная скорость согласно стандарту 20000 Бод. Однако современное оборудование может работать значительно быстрее. Не имеет значения на сколько быстрое (медленное) ваше соединение - максимальное число чтений за секунду можно установить с помощью используемого программного обеспечения.

Обнаружения ошибок при передаче данных. Чётность (Parity) в

RS-232

При передаче по последовательному каналу контроль четности может быть использован для обнаружения ошибок при передаче данных. При использовании контроля четности посылаются сообщения подсчитывающие число единиц в группе бит данных. В зависимости от результата устанавливается бит четности. Приемное устройство также подсчитывает число единиц и затем сверяет бит четности.

Типы четности

Для обеспечения контроля четности компьютер и устройство должны одинаково производить подсчет бита четности. То есть, определиться устанавливать бит при четном (even) или нечетном (odd) числе единиц.

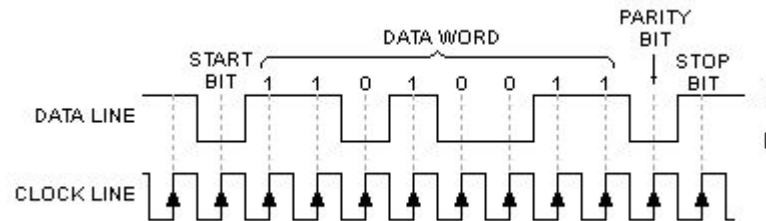
При контроле на четность биты данных и бит четности всегда должны содержать четное число единиц. В противоположном случае соответствует для контроля на нечетность.

Обнаружение ошибок

Проверка на четность - это простейший способ обнаружения ошибок. Он может определить возникновение ошибок в одном бите, но при наличии ошибок в двух битах уже не заметит ошибок. Также такой контроль не отвечает на вопрос какой бит ошибочный.

Другой механизм проверки включает в себя Старт и Стоп биты, циклические проверки на избыточность, которые часто применяются, например, в соединениях Modbus протокола.

Пример. В этом примере показана структура передаваемых данных со синхронизирующим тактовым сигналом. В этом примере используется 8 бит данных, бит четности и стоп бит. Такая структура также обозначается 8E1.



Примечание: Тактовый сигнал - для асинхронной передачи это внутренний сигнал

Старт бит. Сигнальная линия может находиться в двух состояниях: включена и выключена. Линия в состоянии ожидания всегда включена. Когда устройство или компьютер хотят передать данные, они переводят линию в состояние выключено - это установка Старт бита. Биты сразу после Старт бита являются битами данных.

Стоп бит. Стоп бит позволяет устройству или компьютеру произвести синхронизацию при возникновении сбоев. Например, помеха на линии скрыла Старт бит. Период между старт и стоп битами постоянен, согласно значению скорости обмена, числу бит данных и бита четности. Стоп бит всегда включен. Если приемник определяет выключенное состояние, когда должен присутствовать стоп бит, фиксируется появление ошибки.

Установка Стоп бита. Стоп бит не просто один бит минимального интервала времени в конце каждой передачи данных. На компьютерах обычно он эквивалентен **1 или 2** битам, и это должно учитываться программе драйвера. Хотя, 1 стоп бит наиболее общий, выбор 2 бит в худшем случае немного замедлит передачу сообщения.

(Есть возможность установки значения стоп бита равным **1.5**. Это используется при передаче менее 7 битов данных. В этом случае не могут быть переданы символы ASCII, и поэтому значение 1.5 используется редко.)

Преобразование уровней RS-232 в TTL уровень с помощью MAX232

Техническая документация. Семейство микросхем MAX220-MAX249 линейных приемопередатчиков предназначены для интерфейсов EIA/TIA-232E и V.28/V.24, особенно в устройствах, где отсутствуют напряжения $\pm 12\text{В}$.

Микросхема MAX232

Микросхема MAX232 быстро стала промышленным стандартом. Многие разработчики используют ее, несмотря на то, что параметры микросхем с однополярным питанием значительно улучшились со временем.

Конфигурация выводов MAX232: представлена на рис.



MAX232 Pin Nbr.	MAX232 Pin Name	Signal	Voltage	DB9 Pin
7	T2out	RTS	RS-232	8
8	R2in	CTS	RS-232	7
9	R2out	CTS	TTL	n/a
10	T2in	RTS	TTL	n/a
11	T1in	TX	TTL	n/a
12	R1out	RX	TTL	n/a
13	R1in	TX	RS-232	3
14	T1out	RX	RS-232	2
15	GND	GND	0	5

Рис.8 Конфигурация выводов MAX232

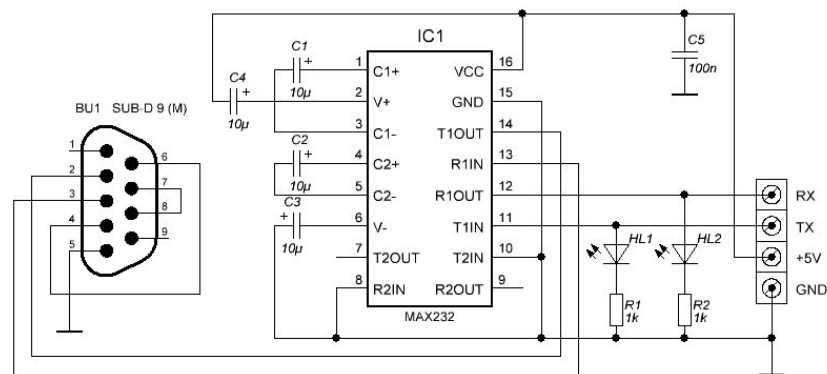


Рис.9 Типовая схема включения MAX232

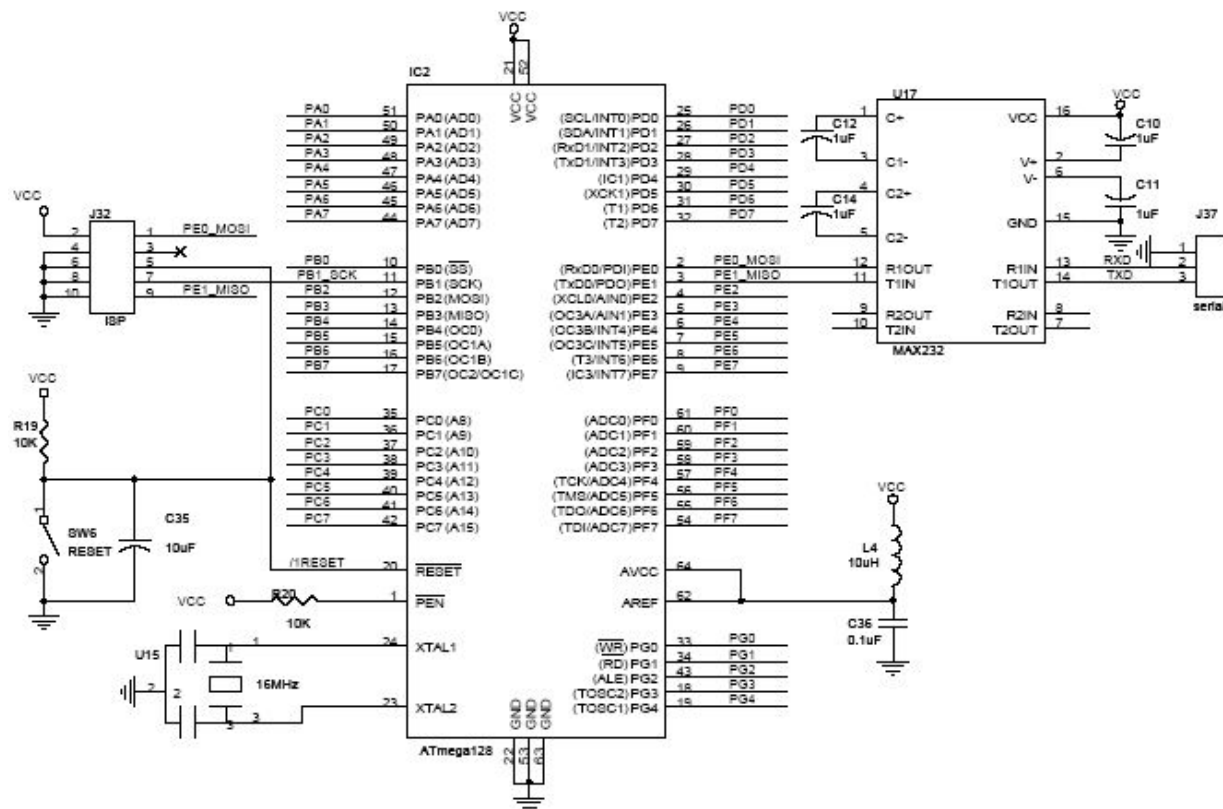


Рис.9 Подключение MAX232 к микроконтроллеру ATmega128

