Module 5

Embedded Communications

Lesson 25

Serial Data Communication

Instructional Objectives

After going through this lesson the student would be able to

- Distinguish between serial and parallel data communication
- Explain why a communication protocol is needed
- Distinguish between the RS-232 and other serial communication standards
- Describe how serial communication can be used to interconnect two remote computers using the telephone line

Questions & Answers

Question	**Visual (If any)**	**A**	**B**	**C**	**D**	**Ans.**
The minimum number of lines used in two-way (full-duplex) serial data transmission is		1	2	3	4	C
The digital signals need to be converted to audio tones for transmission through telephone lines because the bandwidth of these lines is low		T	F			A
A DCE transmits its digital output data through the line		TXD	RXD	DTR	DSR	B
Differential signaling is used to reduce the effect of signal attenuation in the transmission line		T	F			B

MAIN TOPICS ….!!

(CLICK ON THE HYPERLINKS BELOW….!!)

(home…..)

Serial Data Communication

Data Communication is one of the most challenging fields today as far as technology development is concerned. Data, essentially meaning information coded in digital form, that is, 0s and 1s, is needed to be sent from one point to the other either directly or through a network.

And when many such systems need to share the same information or different information through the same medium, there arises a need for proper organization (rather, “socialization”) of the whole network of the systems, so that the whole system works in a cohesive fashion.

Therefore, in order for a proper interaction between the data transmitter (the device needing to commence data communication) and the data receiver (the system which has to receive the data sent by a transmitter) there has to be some set of rules or (“protocols”) which all the interested parties must obey.

The requirement above finally paves the way for some ***DATA COMMUNICATION STANDARDS.***

Depending on the requirement of applications, one has to choose the type of communication strategy. There are basically two major classifications, namely SERIAL and PARALLEL, each with its variants. The discussion about serial communication will be undertaken in this lesson.

Any data communication standard comprises

- The protocol.
- Signal/data/port specifications for the devices or additional electronic circuitry involved.

What is Serial Communication?

(home…..)

Serial data communication strategies and, standards are used in situations having a limitation of the number of lines that can be spared for communication. This is the primary mode of transfer in long-distance communication. But it is also the situation in embedded systems where various subsystems share the communication channel and the speed is not a very critical issue.

Standards incorporate both the software and hardware aspects of the system while ***buses*** mainly define the cable characteristics for the same communication type.

Serial data communication is the most common ***low-level*** protocol for communicating between two or more devices. Normally, one device is a computer, while the other device can be a modem, a printer, another computer, or a scientific instrument such as an oscilloscope or a function generator.

As the name suggests, the serial port sends and receives bytes of information, rather characters *(used in the other modes of communication)*, in a serial fashion - one bit at a time. These bytes are transmitted using either a ***binary (numerical) format*** or *a* ***text format***.

All the data communication systems follow some specific set of standards defined for their communication capabilities so that the systems are not Vendor specific but for each system the user has the advantage of selecting the device and interface according to his own choice of make and range.

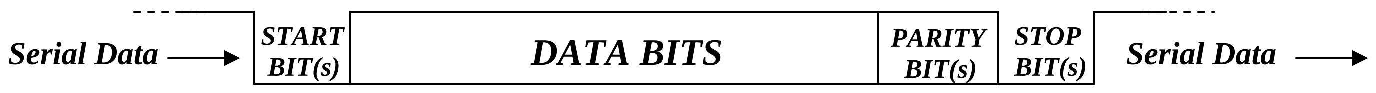
The most common serial communication system protocols can be studied under the following categories: ***Asynchronous, Synchronous*** *and* ***Bit-Synchronous*** *communication standards.*

Asynchronous Communication and Standards ***(home…..)***

The Protocol

- This protocol allows bits of information to be transmitted between two devices at an arbitrary point of time.
- The protocol defines that the data, more appropriately a "character" is sent as "frames" which in turn is a collection of bits.
- The start of a frame is identified according to a ***START*** bit(s) and a ***STOP*** bit(s) identifies the end of data frame. Thus, the ***START*** and the ***STOP*** bits are part of the frame being sent or received.
- The protocol assumes that both the transmitter and the receiver are configured in the same way, i.e., follow the same definitions for the start, stop and the actual data bits.
- Both devices, namely, the transmitter and the receiver, need to communicate at an agreed upon data rate ***(baud rate)*** such as **19,200 KB/s or 115,200 KB/s.**
- This protocol has been in use for 15 years and is used to connect PC peripherals such as ***modems*** and the applications include the classic ***Internet dial-up*** modem systems.
- Asynchronous systems allow a number of variations including the number of ***bits in a character*** (5, 6, 7 or 8 bits), the number of ***stops bits*** used (1, 1.5 or 2) and an optional ***parity bit***. Today the most common standard has 8 bit characters, with 1 stop bit and no parity and this is frequently abbreviated as ***'8-1-n'***. A single ***8-bit*** character, therefore, consists of ***10 bits on the line***, i.e., One ***Start*** bit, Eight ***Data*** bits and One ***Stop*** bit (as shown in the figure below).
- Most important observation here is that the individual characters are ***framed*** (unlike all the other standards of serial communication) and ***NO CLOCK*** data is communicated between the two ends.

The Typical Data Format *(known as "FRAME")* for Asynchronous Communication



Interface Specifications for Asynchronous Serial Data Communication

The ***serial port interface*** for connecting two devices is specified by the **TIA** (*Telecommunications Industry Association*) / **EIA-232C** *(Electronic Industries Alliance)* standard published by the Telecommunications Industry Association; both the physical and electrical characteristics of the interfaces have been detailed in these publications.

RS-232, ***RS-422***, ***RS-423*** ***and*** ***RS-485*** are each a recommended standard (RS-XXX) of the Electronic Industry Association (EIA) for asynchronous serial communication and have more recently been rebranded as ***EIA-232, EIA-422, EIA-423 and EIA-485.***

It must be mentioned here that, although, some of the more advanced standards for serial communication like the USB and FIREWIRE are being popularized these days to fill the gap for high-speed, relatively short-run, heavy-data-handling applications, but still, the above four satisfy the needs of all those high-speed and longer run applications found most often in industrial settings for plant-wide security and equipment networking.

RS-232, 423, 422 and 485 specify the ***communication system characteristics*** of the hardware such as ***voltage levels, terminating resistances, cable lengths, etc***. ***The standards, however, say nothing about the software protocol or how data is framed, addressed, checked for errors or interpreted***

THE RS-232 ***(home…..)***

This is the original serial port interface "standard" and it stands for ***"Recommended Standard Number 232"*** or more appropriately *EIA Recommended Standard 232* is the oldest and the most popular serial communication standard. It was first introduced in 1962 to help ensure connectivity and compatibility across manufacturers for simple serial data communications.

Applications ***(home…..)***

- Peripheral connectivity for PCs *(the* ***PC COM*** *port hardware)*, which can range beyond modems and printers to many different handheld devices and modern scientific instruments.

All the various characteristics and definitions pertaining to this standard can be summarized according to:

- The maximum bit transfer rate capability and cable length.
- Communication Technique: names, electrical characteristics and functions of signals.
- The mechanical connections and pin assignments.

The Standard

Maximum Bit Transfer Rate, Signal Voltages and Cable Length

- RS-232's capabilities range from the original slow data rate of up to 20 kbps to over 1 Mbps for some of the modern applications.
- RS-232 is mainly intended for short cable runs, or local data transfers in a range up to ***50 feet*** maximum, but it must be mentioned here that it also depends on the ***Baud Rate***.

- It is a robust interface with speeds to 115,200 baud, and
- It can ***withstand a short circuit*** between any 2 pins.
- It can handle signal voltages as high / low as ±15 volts.

Signal States and the Communication Technique

(home…..)

Signals can be in either an ***active*** state or an ***inactive*** state. RS232 is an Active LOW voltage driven interface where:

ACTIVE STATE: An active state corresponds to the binary value 1. An active signal state can also be indicated as ***logic "1", "on", "true", or a "mark".***

INACTIVE STATE: An inactive signal state is stated as ***logic "0", "off", "false", or a "space".***

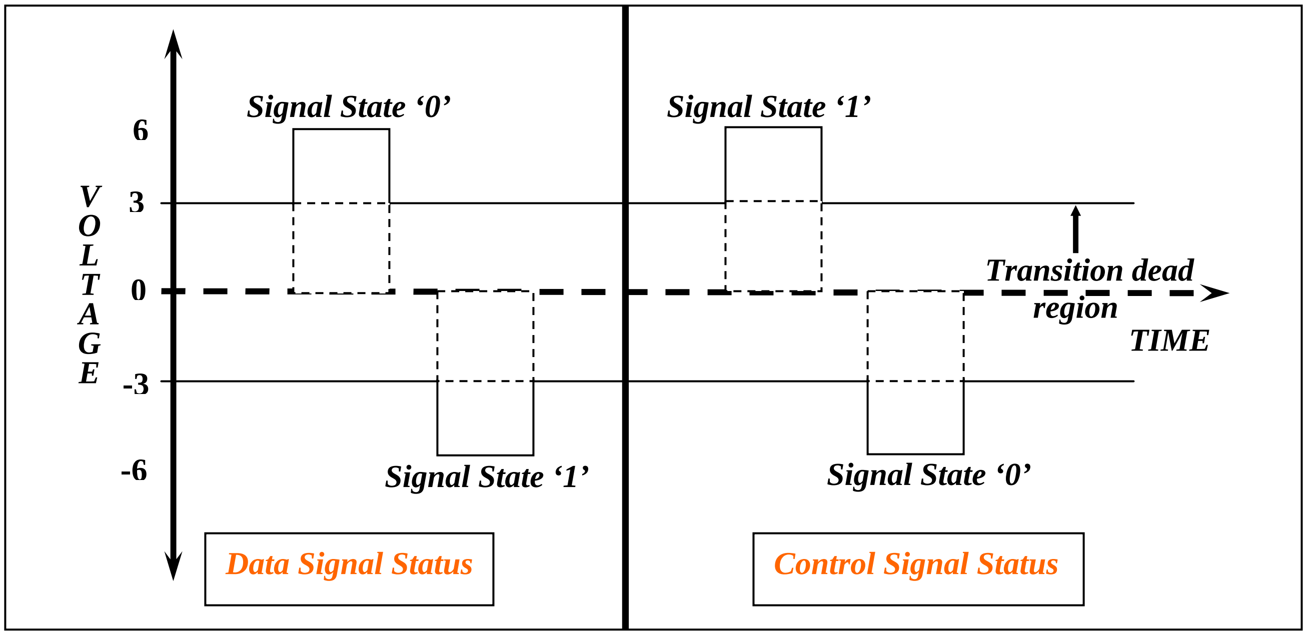
- For data signals, the ***"true"*** state occurs when the received signal voltage is more negative than ***-3 volts***, while the ***"false"*** state occurs for voltages more positive than ***3 volts.***
- For control signals, the ***"true"*** state occurs when the received signal voltage is more positive than ***3 volts***, while the ***"false"*** state occurs for voltages more negative than ***-3 volts***.

Transition or "Dead Area"

Signal voltage region in the range >-3.0V and < +3.0V is regarded as the 'dead area' and allows for absorption of noise. This same region is considered a transition region, and the ***signal state is undefined.***

To bring the signal to the "true" state, the controlling device ***unasserts*** (or *lowers*) the value for data pins and ***asserts*** (or *raises*) the value for control pins. Conversely, to bring the signal to the "false" state, the controlling device asserts the value for data pins and unasserts the value for control pins. The "true" and "false" states for a data signal and for a control signal are as shown below.

The Communication Technique

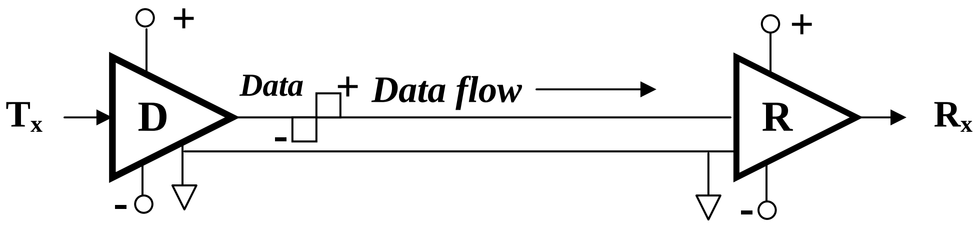


A factor that limits the distance of reliable data transfer using RS-232 is the signaling technique that it uses.

- This interface is ***"single-ended"*** meaning that communication occurs over a ***SINGLE WIRE referenced to GROUND***, the ground wire serving as a second wire. Over that single wire, ***marks and spaces*** are created.
- While this is very adequate for slower applications, it is not suitable for faster and longer applications.

The communication technique

- RS-232 is designed for a ***unidirectional half-duplex communications mode***. That simply means that a transmitter (driver) is feeding the data to a receiver over a copper line. The data always follows the direction from driver to receiver over that line. If ***return transmission*** is desired, another set of driver- receiver pair and separate wires are needed. In other words, if ***bi-directional or full-duplex capabilities*** are needed, two separate communications paths are required.



RS-232 – Single-Ended, Unidirectional, Half Duplex

Disadvantage

(home…..)

Being a ***single-ended system*** it is more susceptible to induced noise, ground loops and ground shifts, a ground at one end not the same potential as at the other end of the cable e.g. in applications under the proximity of heavy electrical installations and machineries But these vulnerabilities at very high data rates and for those applications a different standard, like the RS-422 etc., is required which have been explained further.

Some Modern Perspectives/Advantages

Most applications for RS-232 today are for data connectivity between portable handheld devices and a PC. Some of the differences between the modern RS-232 integrals from the older versions are:

- Such devices require that the RS-232 IC to be very small, have low current drain, operate from a +3 to +5-V supply.
- They provide **ESD** protection on all transmit and receive pins. For example, some RS-232 interfaces have specifically been designed ***for handheld devices*** and support ***data rates greater than 250 kbps***, can operate down to ***+2.7 V***.
- They can automatically go into a ***standby mode*** drawing very small currents of the order of only ***150 nA*** when not in use, provide ***15 kV ESD*** protection on data pins and are in the ***near-chip-scale 5 X 5 mm quad flat no-lead package.***

Nevertheless, for portable and handheld applications the older RS-232 is still the most popular one.

(home…..)

RS-422 and RS-423 (EIA Recommended Standard 422 and 423)

These were designed, specifically; to overcome the distance and speed limitations of RS-232.Although they are similar to the more advanced RS-232C, but can accommodate higher baud rates and longer cable lengths and, accommodate multiple receivers.

The Standard

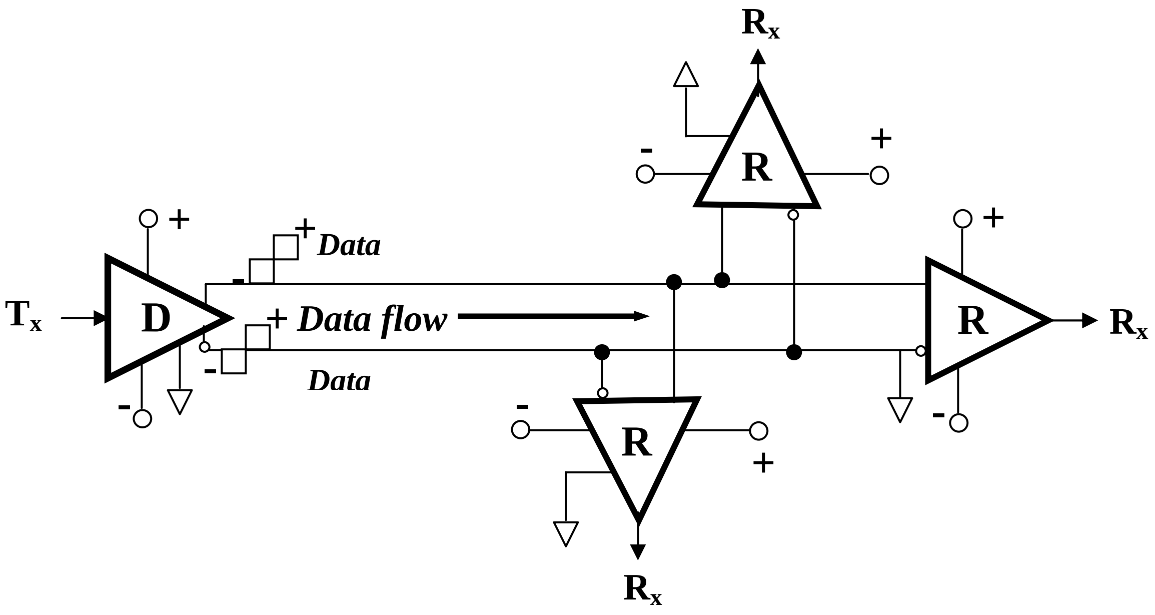
(home…..)

Maximum Bit Transfer Rate, Signal Voltages and Cable Length

- For both of these standards the data lines can be up to ***4,000 feet*** with a data rate around ***100 kbps***.
- The ***maximum data rate*** is around ***10 Mbps for short runs***, trading off distance for speed.
- The maximum signal voltage levels are ***±6 volts***.
- The signaling technique for the RS-422 and RS-423 is mainly responsible for there superiority over RS-232 in terms of speed and length of transmission as explained in the next subsection.

Communication Technique

- The flair of this standard lies in its capability in tolerating the ground voltage differences between sender and receiver. Ground voltage differences can occur in ***electrically noisy*** environments where ***heavy electrical machinery*** is operating.
- The criterion here is the ***differential-data communication technique***, also referred to as ***balanced-differential signaling***. In this, the driver uses two wires over which the signal is transmitted. However, each wire is driven and floating separate from ground, meaning, neither is grounded and in this respect this system is different to the single-ended systems. Correspondingly, the receiver has two inputs, each floating above ground and electrically balanced with the other when no data is being transmitted. Data on the line causes a desired electrical imbalance, which is recognized and amplified by the receiver. The ***common-mode signals***, such as induced electrical noise on the lines caused from machinery or radio transmissions, are, for the most part, canceled by the receiver. That is because the induced noise is identical on each wire and the receiver ***inverts*** the signal on one wire to place it ***out of phase*** with the other causing a ***subtraction*** to occur which results in a ***Zero difference.*** Thus, noise picked up by the long data lines is eliminated at the receiver and does not interfere with data transfer. Also, because the line is balanced and separate from ground, there is no problem associated with ground shifts or ground loops.



RS-422 – Differential Signaling, Unidirectional, Half Duplex, Multi-drop

- It may be mentioned here to avoid any ambiguity in understanding the RS-422 and the RS-423 standards, that, the standard RS-423 is an advanced counterpart of RS-422 which has been designed to tolerate the ground voltage differences between the sender and the receiver for the more advanced version of RS-232, that is, the RS-232C.
- Unlike RS-232, an RS-422 driver can service up to ***10 receivers*** on the ***same line (bus)***. This is often referred to as a ***half-duplex single-source multi-drop network***, (not to be confused with ***multi-point networks*** associated with RS-485), this will be explained further in conjugation with RS-485.

- Like RS-232, however, RS-422 is still ***half-duplex*** one-way data communications over a two-wire line. If ***bi-directional or full-duplex*** operation is desired, another set of driver, receiver(s) and two-wire line is needed. In which case, RS-485 is worth considering.

Applications

This fits well in process control applications in which instructions are sent out to **many *actuators or responders*. Ground voltage differences can occur in electrically noisy environments where heavy electrical machinery is operating.**

RS-485

This is an improved RS-422 with the capability of connecting a number of devices (transceivers) on ***one serial bus to form a network.***

The Standard

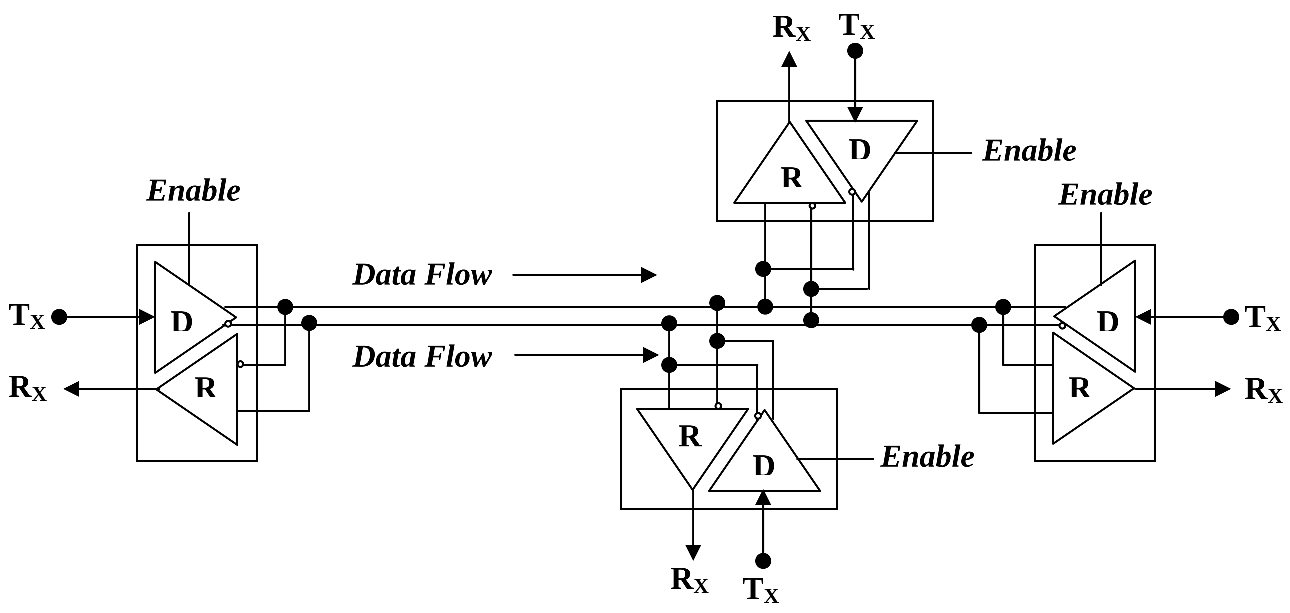
Maximum Bit Transfer Rate, Signal Voltages and Cable Length

- Such a network can have a "daisy chain" topology where each device is connected to two other devices except for the devices on the ends.
- Only one device may drive data onto the bus at a time. The standard does not specify the rules for deciding who transmits and when on such a network. That solely depends upon the system designer to define.
- Variable data rates are available for this standards but the standard max. data rate is 10 Mbps, however ,some manufacturers do offer up to double the standard range i.e. around 20 Mbps,but of course, it is at the expense of cable width.
- It can connect upto 32 drivers and receivers in fully differential mode similar to the RS – 422.

Communication Technique

(home…)

- EIA Recommended Standard 485 is designed to provide ***bi-directional half-duplex multi-point data communications over a single two-wire bus***.
- Like RS-232 and RS-422, full-duplex operation is possible using a ***four-wire, two-bus network but the RS-485 transceiver ICs*** must have separate transmit and receive pins to accomplish this.
- RS-485 has ***the same distance and data rate specifications as RS-422*** and uses differential signaling but, ***unlike RS-422, allows multiple drivers on the same bus***. As depicted in the Figure below, each node on the bus can include both a driver and receiver forming a multi-point star network. Each driver at each node remains in a ***disabled high-impedance state*** until called upon to transmit. This is different than drivers made for RS-422 where there is only one driver and it is always enabled and cannot be disabled.
- With automatic repeaters and tri-state drivers the 32-node limit can be greatly exceeded. In fact, the ANSI-based SCSI-2 and SCSI-3 bus specifications use RS-485 for the physical (hardware) layer.



RS-485 – Differential Signaling, Bi-directional, Half Duplex, Multi-point

Advantages

- Among all of the asynchronous standards mentioned above this standard offers the maximum data rate.
- Apart from that special hardware for avoiding bus contention and ,
- A higher receiver input impedance with lower Driver load impedances are its other assets.

(home…..)

Differences between the various standards at a glance

All together the important electrical and mechanical characteristics for application purposes may be classified and summarized according to the table below.

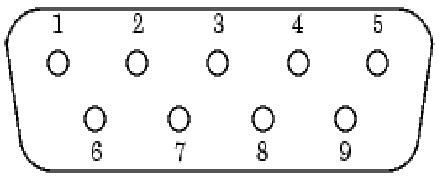
	RS-232	**RS-422/423**	**RS-485**
Signaling Technique	Single-Ended (Unbalanced)	Differential (Balanced)	Differential (Balanced)
Drivers and Receivers on Bus	1 Driver 1 Receiver	1 Driver 10 Receivers	32 Drivers 32 Receivers
Maximum Cable Length	50 feet	4000 feet	4000 feet
Original Standard Maximum Data Rate	20 kbps	10 Mbps down to 100 kbps	10 Mbps down to 100 kbps
Minimum Loaded Driver Output Voltage Levels	+/-5.0 V	+/-2.0 V	+/-1.5 V

Driver Load Impedance	3 to 7 k	100	54
Receiver Input Impedance	3 to 7 k	4 k or greater	12 k or greater

(home…..)

Interfacing of Peripherals Involving the Rs-232 Asynchronous Communication Standards

The RS-232 standard defines the two devices connected with a serial cable as the ***Data Terminal Equipment*** *(DTE)* and ***Data Circuit-Terminating Equipment*** *(DCE*). This terminology reflects the RS-232 origin as a standard for communication between a ***computer terminal*** and a ***modem***. Primary communication is accomplished using three pins: the ***Transmit Data*** (TD) pin, the ***Receive Data***(RD) pin, and the ***Ground pin*** (not shown). Other pins are available for ***data flow control***. The serial port pins and the signal assignments for a typical asynchronous serial communication can be shown in the scheme for a 9-pin male connector ***(DB9) on the DTE*** as under:



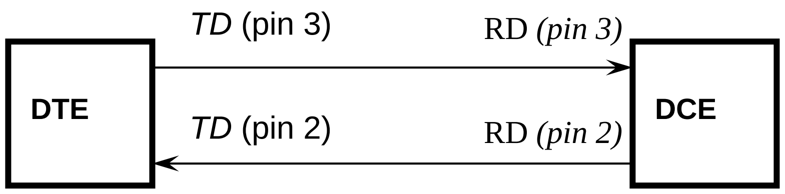
The DB9 male connector

Serial Port Pin and Signal Assignments

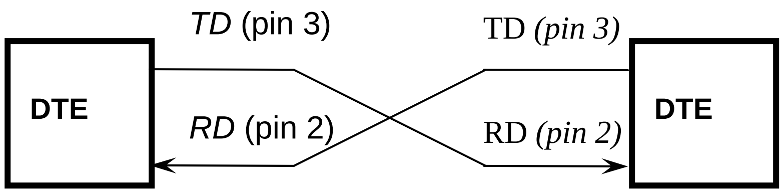
Pin	***Label***	***Signal Name***	***Signal Type***
1	CD	*Carrier Detect*	Control
2	RD	*Received Data*	Data
3	TD	*Transmitted Data*	Data
4	DTR	*Data Terminal Ready*	Control
5	GND	*Signal Ground*	Ground
6	DSR	*Data Set Ready*	Control
7	RTS	*Request to Send*	Control
8	CTS	*Clear to Send*	Control
9	RI	*Ring Indicator*	Control

(The RS-232 standard can be referred for a description of the signals and pin assignments used for a 25-pin connector)

Because RS-232 mainly involves connecting a DTE to a DCE, the pin assignments are defined such that ***straight-through cabling*** is used, where pin 1 is connected to pin 1, pin 2 is connected to pin 2, and so on. A DTE to DCE serial connection using the ***Transmit Data (TD)*** pin and the ***Receive Data (RD)*** pin is shown below.



Connecting two DTE's or two DCE's using a straight serial cable, means that the TD pin on each device are connected to each other, and the RD pin on each device are connected to each other. Therefore, to connect two like devices, a ***null modem*** cable has to be used. As shown below, null modem cables crosses the transmit and receive lines in the cable.



Serial ports consist of two signal types: data signals and control signals. To support these signal types, as well as the signal ground, the RS-232 standard defines a 25-pin connection. However, most PC's and UNIX platforms use a 9-pin connection. In fact, only three pins are required for serial port communications: one for receiving data, one for transmitting data, and one for the signal ground.

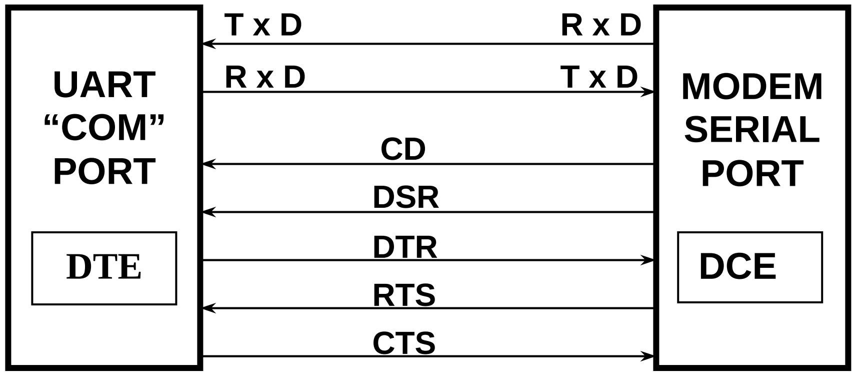
Throughout this discussion computer is considered a DTE, while peripheral devices such as modems and printers are considered DCE's. Note that many scientific instruments function as DTE's.

The term "data set" is synonymous with "modem" or "device," while the term "data terminal" is synonymous with "computer."

(Detail PC – PC communication….)

(home…..)

The schematic for a connection between the PC UART port and the Modem serial port is as shown below:



Note: The serial port pin and signal assignments are with respect to the DTE. For example, data is transmitted from the TD pin of the DTE to the RD pin of the DCE.

The Data Pins

Most serial port devices support ***full-duplex*** communication meaning that they can send and receive data at the same time. Therefore, separate pins are used for transmitting and receiving data. For these devices, the TD, RD, and GND pins are used. However, some types of serial port devices support only one-way or ***half-duplex*** communications. For these devices, only the TD and GND pins are used. In the course of explanation, it is assumed that a full-duplex serial port is connected to the DCE.
The TD pin carries data transmitted by a DTE to a DCE. The RD pin carries data that is received by a DTE from a DCE.

The Control Pins

9-pin serial ports provide several control pins whose functions are to:

- Signal the presence of connected devices
- Control the flow of data

The control pins include RTS and CTS, DTR and DSR, CD, and RI.

The RTS and CTS Pins

The RTS and CTS pins are used to signal whether the devices are ready to send or receive data. This type of data flow control - called hardware handshaking - is used to prevent data loss during transmission. When enabled for both the DTE and DCE, hardware handshaking using RTS and CTS follows these steps:

1. The DTE asserts the RTS pin to instruct the DCE that it is ready to receive data.
2. The DCE asserts the CTS pin indicating that it is clear to send data over the TD pin. If data can no longer be sent, the CTS pin is unasserted.
3. The data is transmitted to the DTE over the TD pin. If data can no longer be accepted, the RTS pin is unasserted by the DTE and the data transmission is stopped.

The DTR and DSR Pins

Many devices use the DSR and DTR pins to signal if they are connected and powered. Signaling the presence of connected devices using DTR and DSR follows these steps:

1. The DTE asserts the DTR pin to request that the DCE connect to the communication line.
2. The DCE asserts the DSR pin to indicate it's connected.
3. DCE unasserts the DSR pin when it's disconnected from the communication line.

The DTR and DSR pins were originally designed to provide an alternative method of hardware handshaking. However, the RTS and CTS pins are usually used in this way, and not the DSR and DTR pins. However, you should refer to your device documentation to determine its specific pin behavior.

The CD and RI Pins

The CD and RI pins are typically used to indicate the presence of certain signals during modem-modem connections.

CD is used by a modem to signal that it has made a connection with another modem, or has detected a carrier tone. CD is asserted when the DCE is receiving a signal of a suitable frequency. CD is unasserted if the DCE is not receiving a suitable signal.

RI is used to indicate the presence of an audible ringing signal. RI is asserted when the DCE is receiving a ringing signal. RI is unasserted when the DCE is not receiving a ringing signal (for example, it's between rings).

A Practical Example: PC-PC Communication

(home…..)

PROBLEM: Suppose one PC needs to send data to another computer located far away from its vicinity. Now, the actual data is in the parallel form, it needs to be converted into its serial counterpart. This is done by a Parallel-in-Serial-out Shift register and a Serial-in-Parallel-out Shift register (some electronic component).

It has to be made sure that the transmitter must not send the data at a rate faster than with which the receiver can receive it. This is done by introducing some handshaking signals or circuitry in conjugation with the actual system.

For very short distances, devices like UART(Universal Asynchronous Receiver Transmitter: IN8250 from National Semiconductors Corporation) and USART (Universal Synchronous Asynchronous Receiver Transmitter; Intel 8251A from Intel Corporation.) incorporate the essential circuitry for handling this serial communication with handshaking.

For long distances ***Telephone lines (switched lines)*** are more practically feasible because of there pre-availability.

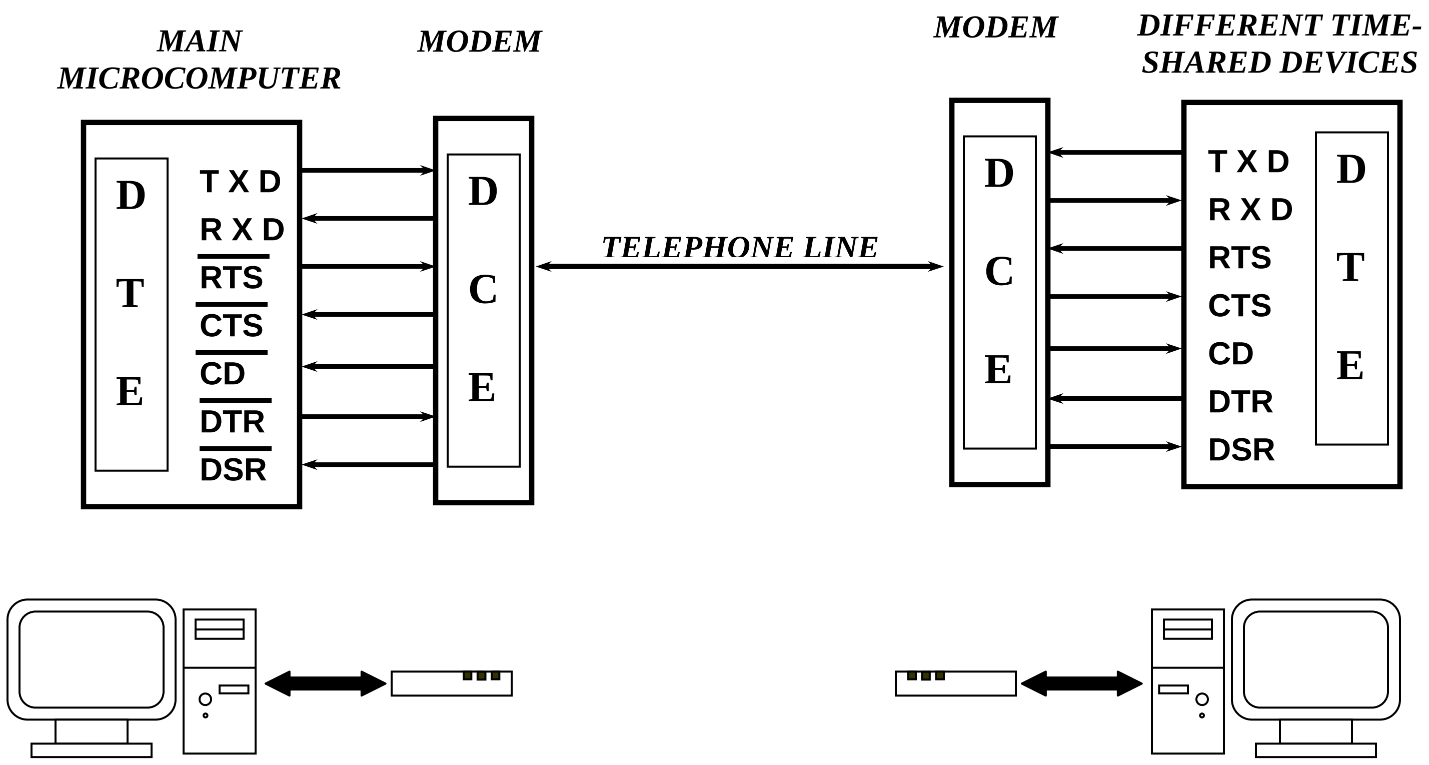
ONE COMPLICATION: …BANDWIDTH is only 300 – 3000Hz.

REMEDY: Convert the digital signal to audio tones. The device, which is used to do this conversion and vice-versa, is known as a **MODEM.**

But how all the above Principles are Applied in Practice?

Consider the control room of a steel plant where one main computer is time-sharing and communicating data to and fro with some other computers or I/O modules in a DCS or SCADA hierarchy.

In the simplest way the communication hardware-software can be represented in a top-level block diagram as follows:



A TYPICAL DIGITAL TRANSMISSION SYSTEM

Overall Procedure of Communication……………………….

(Note: This is actually the initialization and handshaking description for a typical UART, the Intel 8251A)…… (for more details click the box here)

To start with, it should be mentioned that the signals alongside the arrowheads represent the minimum number of necessary signals for the execution of a typical communication standard or a protocol; being elaborated later. These signals occur when the main control terminal wants to send some control signal to the end device or if the end device wants to send some data, say an alarm or some process output, to the main controller.

Both the main microcomputer and the end-device or the time-shared device can be referred to as terminals.

Whenever a terminal is switched on it first performs a self-diagnostic test, in which it checks itself and if it finds that its integrity is fully justified it asserts the DTR ***(data-terminal ready)*** signal low. As the modem senses it getting low, it understands that the terminal is ready.

The modem then "replies" the terminal by asserting DSR ***(data-set ready)*** signal low. Here the direction of the arrows is of prime importance and must be remembered to get the full understandability of the whole procedure.

If the terminal is actually having some valuable data to convey to the end-terminal it will assert the RTS ***(request-to-send)*** signal low back to the modem and, in turn, the modem will assert the CD ***(carrier-detect)*** signal to the terminal indicating as if now it has justified the connection with the terminal computer.

But it may be possible that the modem may not be fully ready to transmit the actual data to the telephone, this may be because of its buffer saturation and several other reasons. When the modem is fully ready to send the data along the telephone line it will assert the CTS (Clear-to-send) signal back to the terminal.

The terminal then starts sending the serial data to the modem and the modem. When the terminal gets exhausted of the data it asserts the RTS signal low indicating the modem that it has not got any more data to be sent. The modem in turn unasserts its CTS signal and stops transmitting.

The same way initialization and the handshaking processes are executed at the other end. Therefore, it must be noted here that the very important aspect of data communication is the definition of the handshaking signals defined for transferring serial data to and from the modem.

Current loops

(home…..)

Current loops are a standard, which are used widely in process automation. 20 mA are wirely used for transmitting serial communication data to programmable process controlling devices. Other widely used standard is 4-20mA current loop, which is used for transmitting analogue measurement signals between the sensor and measurement device.

Serial communication using current loop

(home…..)

In digital communications 20 mA current loop is a standard. The transmitters will only source 20 mA and the receiver will only sink 20 mA. Current loops often use opto-couplers. Here it is the current which matters and not the voltages.

For measurement purposes a small resistance, say of value1k, is connected in series with the receiver/transmitter and the current meter. The current flowing into the receiver indicates the ***scaled*** data, which is actually going inside it. The data transmitted though this kind of interface is usually a standard RS-232 signal just converted to current pulses. Current "on" and "off" the transmission line depends on how the RS-232 circuit distinguishes between the value of currents and in what way it interprets the logic state thus obtained.

4-20 mA current loop

(home…..)

4-20 mA current loop interface is the standard for almost all the process control instruments. This interface works as follows. The sensor is connected to a process controlling equipment, which reads the sensor value and supplies a voltage to the loop where the sensor is connected and reads the amount of current it takes. The typical supply voltage for this arrangement is around ***12-24*** *Volts* through a resistor and the measured output is the voltage drop across that resistor converted into its current counterpart.

The current loop is designed so that a sensor takes 4 mA current when it is at its minimum value and 20 mA when it is in its maximum value.

Because the sensor will always pass at least 4 mA current and there is usually a voltage drop of many volts over the sensor, many sensor types can be made to be powered from only that loop current.