

Data transmission – Transmission modes

Transmission modes

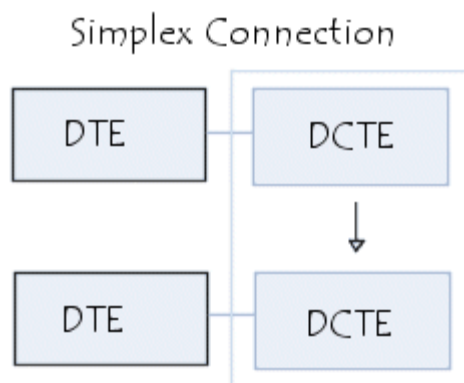
A given transmission on a communications channel between two machines can occur in several different ways. The transmission is characterised by:

- the direction of the exchanges
- the transmission mode: the number of bits sent simultaneously
- synchronization between the transmitter and receiver

Simplex, half-duplex and full-duplex connections

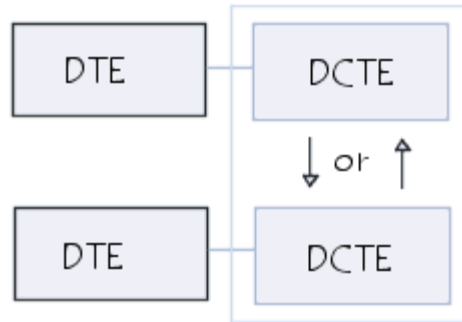
There are 3 different transmission modes characterised according to the direction of the exchanges:

- **A simplex connection** is a connection in which the data flows in only one direction, from the transmitter to the receiver. This type of connection is useful if the data do not need to flow in both directions (for example, from your computer to the printer or from the mouse to your computer...).



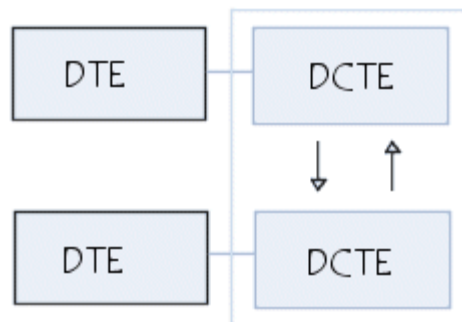
- **A half-duplex connection** (sometimes called an *alternating connection* or *semi-duplex*) is a connection in which the data flows in one direction or the other, but not both at the same time. With this type of connection, each end of the connection transmits in turn. This type of connection makes it possible to have bidirectional communications using the full capacity of the line.

Half-duplex connection



- A **full-duplex connection** is a connection in which the data flow in both directions simultaneously. Each end of the line can thus transmit and receive at the same time, which means that the bandwidth is divided in two for each direction of data transmission if the same transmission medium is used for both directions of transmission.

Full-duplex connection



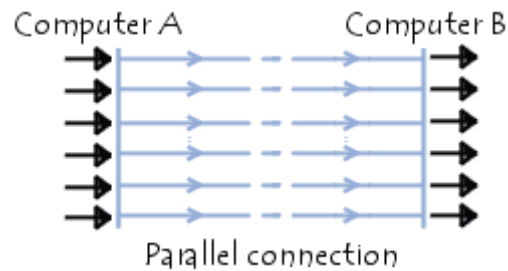
Serial and parallel transmission

The **transmission mode** refers to the number of elementary units of information (bits) that can be simultaneously translated by the communications channel. In fact, processors (and therefore computers in general) never process (in the case of recent processors) a single bit at a time; generally they are able to process several (most of the time it is 8: one byte), and for this reason the basic connections on a computer are parallel connections.

Parallel connection

Parallel connection means simultaneous transmission of N bits. These bits are sent simultaneously over N different channels (a channel being, for example, a *wire*, a cable

or any other physical medium). The parallel connection on PC-type computers generally requires 10 wires.



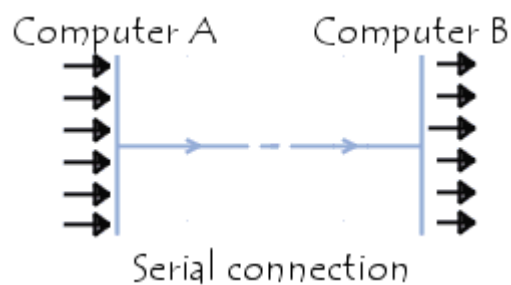
These channels may be:

- N physical lines: in which case each bit is sent on a physical line (which is why parallel cables are made up of several wires in a ribbon cable)
- one physical line divided into several sub-channels by dividing up the bandwidth. In this case, each bit is sent at a different frequency...

Since the conductive wires are close to each other in the ribbon cable, interference can occur (particularly at high speeds) and degrade the signal quality...

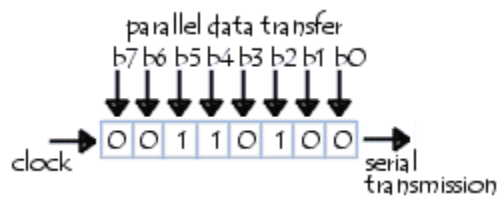
Serial connection

In a serial connection, the data are sent one bit at a time over the transmission channel. However, since most processors process data in parallel, the transmitter needs to transform incoming parallel data into serial data and the receiver needs to do the opposite.

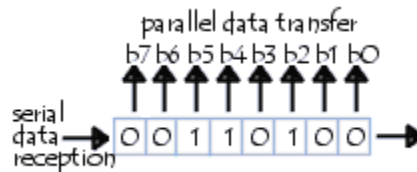


These operations are performed by a communications controller (normally a *UART (Universal Asynchronous Receiver Transmitter)* chip). The communications controller works in the following manner:

- The **parallel–serial transformation** is performed using a shift register. The shift register, working together with a clock, will shift the register (containing all of the data presented in parallel) by one position to the left, and then transmit the most significant bit (the leftmost one) and so on:



- The **serial–parallel transformation** is done in almost the same way using a shift register. The shift register shifts the register by one position to the left each time a bit is received, and then transmits the entire register in parallel when it is full:



Synchronous and asynchronous transmission

Given the problems that arise with a parallel-type connection, serial connections are normally used. However, since a single wire transports the information, the problem is how to synchronize the transmitter and receiver, in other words, the receiver can not necessarily distinguish the characters (or more generally the bit sequences) because the bits are sent one after the other. There are two types of transmission that address this problem:

- An **asynchronous connection**, in which each character is sent at irregular intervals in time (for example a user sending characters entered at the keyboard in real time). So, for example, imagine that a single bit is transmitted during a long period of silence... the receiver will not be able to know if this is 00010000, 10000000 or 00000100...

To remedy this problem, each character is preceded by some information indicating the start of character transmission (the transmission start information is called a *START bit*) and ends by sending end-of-transmission information (called *STOP bit*, there may even be several STOP bits).

- In a **synchronous connection**, the transmitter and receiver are paced by the same clock. The receiver continuously receives (even when no bits are transmitted) the information at the same rate the transmitter send it. This is why the transmitter and receiver are paced at the same speed. In addition, supplementary information is inserted to guarantee that there are no errors during transmission.

During synchronous transmission, the bits are sent successively with no separation between each character, so it is necessary to insert synchronization elements; this is called **character-level synchronization**.

The main disadvantage of synchronous transmission is recognising the data at the receiver, as there may be differences between the transmitter and receiver clocks. That is why each data transmission must be sustained long enough for the receiver to distinguish it. As a result, the transmission speed can not be very high in a synchronous link.

Source: <http://en.kioskea.net/contents/701-data-transmission-transmission-modes>