

## Digital Transmission

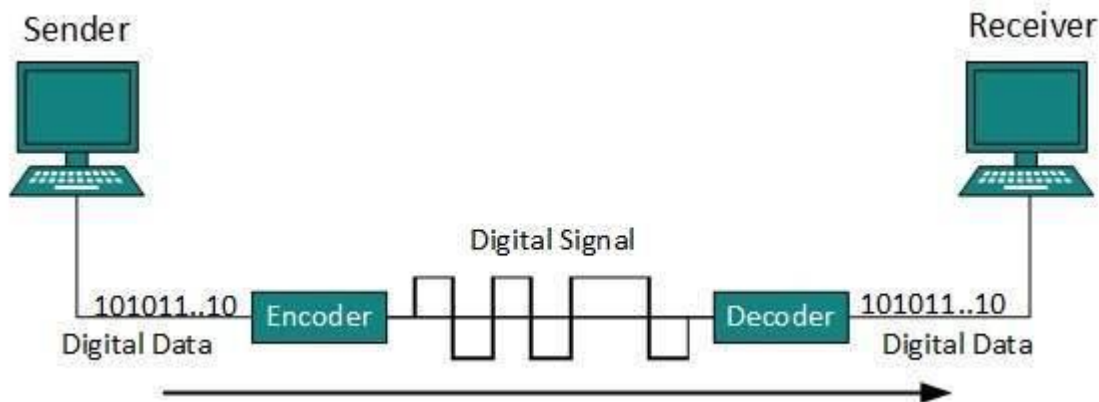
Data or information can be stored in two ways, analog and digital. For a computer to use the data, it must be in discrete digital form. Similar to data, signals can also be in analog and digital form. To transmit data digitally, it needs to be first converted to digital form.

### Digital-to-Digital Conversion

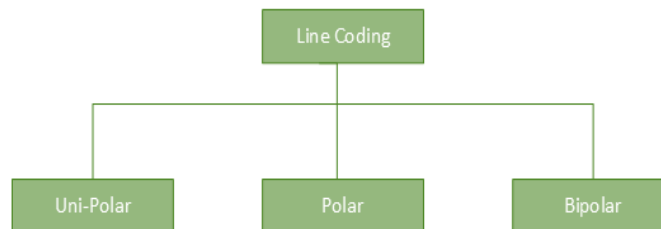
This section explains how to convert digital data into digital signals. It can be done in two ways, line coding and block coding. For all communications, line coding is necessary whereas block coding is optional.

### Line Coding

The process for converting digital data into digital signal is said to be Line Coding. Digital data is found in binary format. It is represented (stored) internally as series of 1s and 0s.

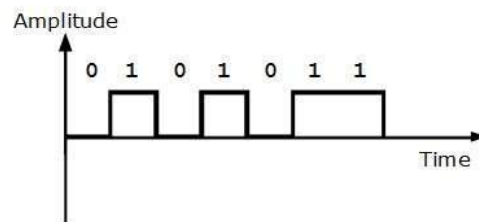


Digital signal is denoted by discrete signal, which represents digital data. There are three types of line coding schemes available:



## A. Uni-polar Encoding

Unipolar encoding schemes use single voltage level to represent data. In this case, to represent binary 1, high voltage is transmitted and to represent 0, no voltage is transmitted. It is also called Unipolar-Non-return-to-zero, because there is no rest condition i.e. it either represents 1 or 0.



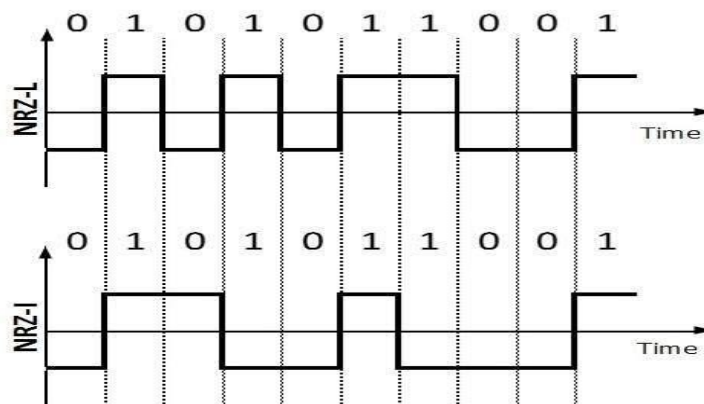
## B. Polar Encoding

Polar encoding scheme uses multiple voltage levels to represent binary values. Polar encodings is available in **four types**:

- **Polar Non-Return to Zero (Polar NRZ)**

It uses two different voltage levels to represent binary values. Generally, positive voltage represents 1 and negative value represents 0. It is also NRZ because there is no rest condition.

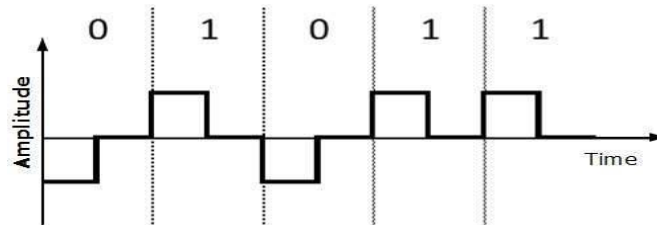
NRZ scheme has two variants: NRZ-L and NRZ-I.



NRZ-L changes voltage level at when a different bit is encountered whereas NRZ-I changes voltage when a 1 is encountered.

- **Return to Zero (RZ)**

Problem with NRZ is that the receiver cannot conclude when a bit ended and when the next bit is started, in case when sender and receiver's clock are not synchronized.



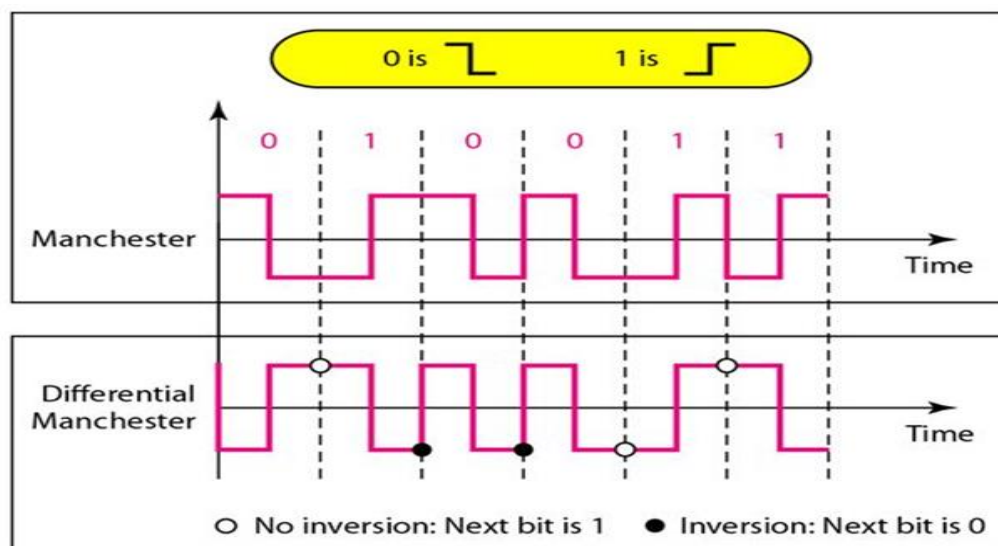
RZ uses three voltage levels, positive voltage to represent 1, negative voltage to represent 0 and zero voltage for none. Signals change during bits not between bits.

- **Manchester**

This encoding scheme is a combination of RZ and NRZ-L. Bit time is divided into two halves. It transits in the middle of the bit and changes phase when a different bit is encountered.

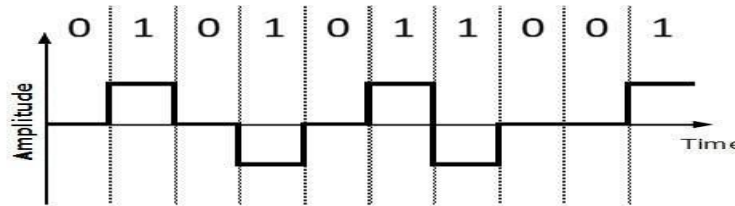
- **Differential Manchester**

This encoding scheme is a combination of RZ and NRZ-I. It also transit at the middle of the bit but changes phase only when 1 is encountered.



### **C. Bipolar Encoding**

Bipolar encoding uses three voltage levels, positive, negative and zero. Zero voltage represents binary 0 and bit 1 is represented by altering positive and negative voltages.



### **Block Coding**

To ensure accuracy of the received data frame redundant bits are used. For example, in even-parity, one parity bit is added to make the count of 1s in the frame even. This way the original number of bits is increased. It is called Block Coding.

Block coding is represented by slash notation, mB/nB. Means, m-bit block is substituted with n-bit block where  $n > m$ . Block coding involves three steps:

- Division,
- Substitution
- Combination.

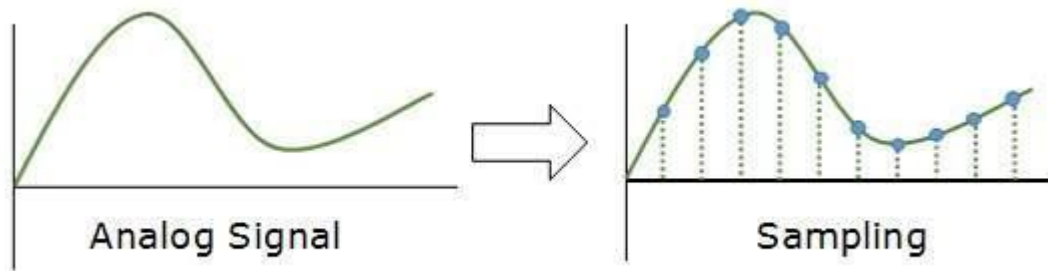
After block coding is done, it is line coded for transmission.

### **Analog-to-Digital Conversion**

Microphones create analog voice and camera creates analog videos, which are treated as analog data. To transmit this analog data over digital signals, we need analog to digital conversion. Analog data is a continuous stream of data in the wave form whereas digital data is discrete. To convert analog wave into digital data, we use **Pulse Code Modulation (PCM)**. PCM is one of the most commonly used method to convert analog data into digital form. It involves three steps:

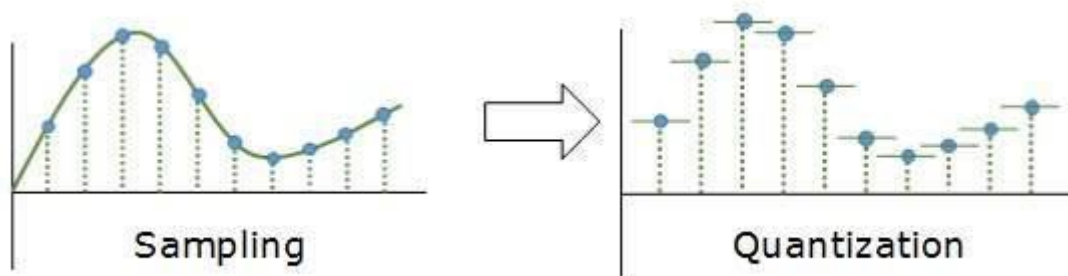
- Sampling
- Quantization
- Encoding.

## Sampling



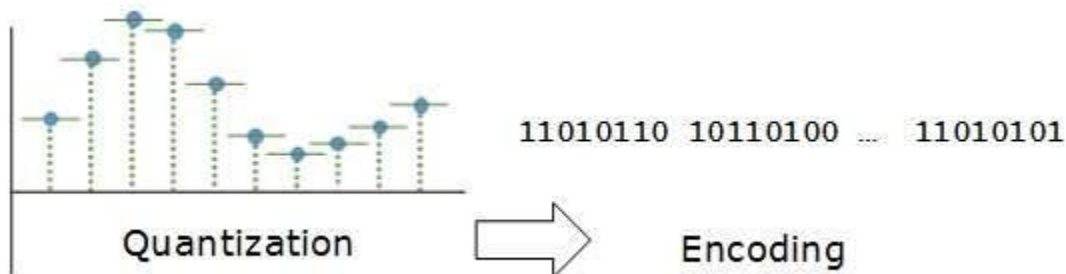
The analog signal is sampled every  $T$  interval. Most important factor in sampling is the rate at which analog signal is sampled. According to Nyquist Theorem, the sampling rate must be at least two times of the highest frequency of the signal.

## Quantization



Sampling yields discrete form of continuous analog signal. Every discrete pattern shows the amplitude of the analog signal at that instance. The quantization is done between the maximum amplitude value and the minimum amplitude value. Quantization is approximation of the instantaneous analog value.

## Encoding

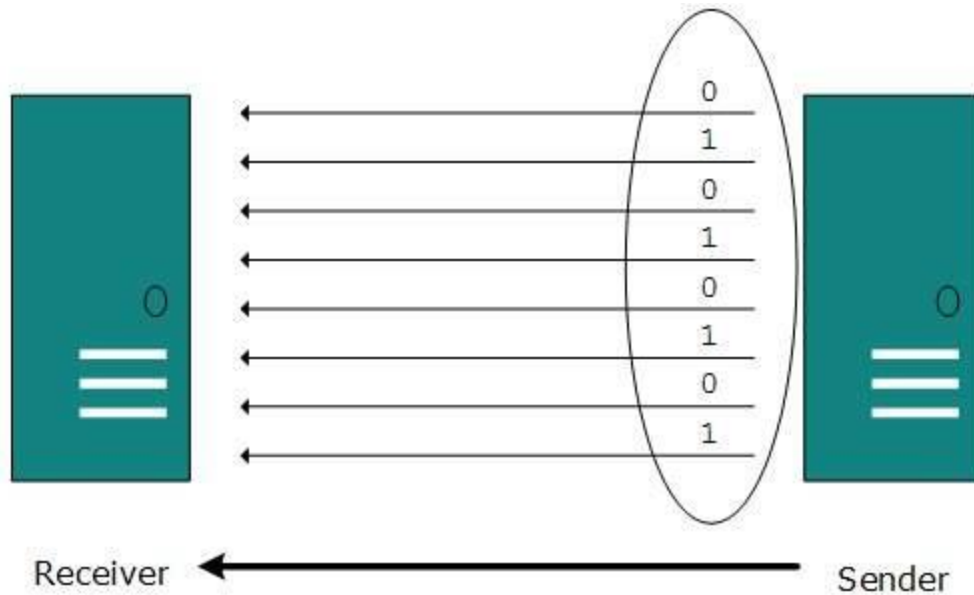


In encoding, each approximated value is then converted into binary format.

## Transmission Modes

The transmission mode decides how data is transmitted between two computers. The binary data in the form of 1s and 0s can be sent in two different modes: Parallel and Serial.

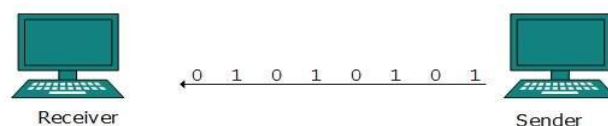
### ➤ Parallel Transmission



The binary bits are organized in-to groups of fixed length. Both sender and receiver are connected in parallel with the equal number of data lines. Both computers distinguish between high order and low order data lines. The sender sends all the bits at once on all lines. Because the data lines are equal to the number of bits in a group or data frame, a complete group of bits (data frame) is sent in one go. Advantage of Parallel transmission is high speed and disadvantage is the cost of wires, as it is equal to the number of bits sent in parallel.

### ➤ Serial Transmission

In serial transmission, bits are sent one after another in a queue manner. Serial transmission requires only one communication channel.

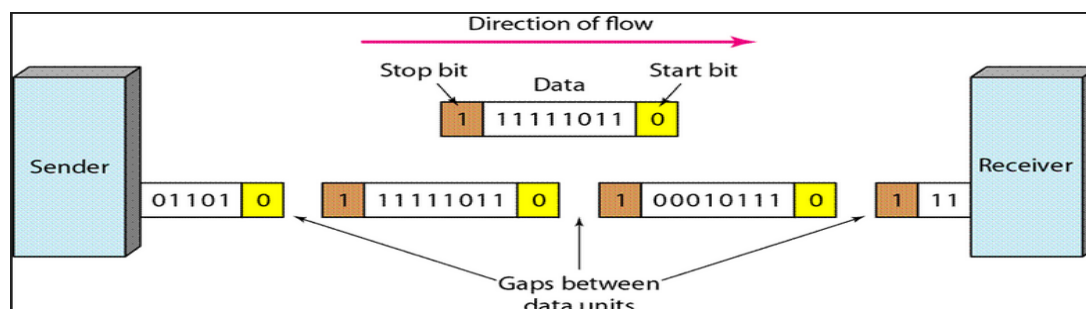


Serial transmission can be either asynchronous or synchronous.

## 1 Asynchronous Serial Transmission

It is named so because there is no importance of timing. Data-bits have specific pattern and they help receiver recognize the start and end data bits. For example, a 0 is prefixed on every data byte and one or more 1s are added at the end.

Two continuous data-frames (bytes) may have a gap between them.



## 2 Synchronous Serial Transmission

Timing in synchronous transmission has importance as there is no mechanism followed to recognize start and end data bits. There is no pattern or prefix/suffix method. Data bits are sent in burst mode without maintaining gap between bytes (8-bits). Single burst of data bits may contain a number of bytes. Therefore, timing becomes very important.

It is up to the receiver to recognize and separate bits into bytes. The advantage of synchronous transmission is high speed, and it has no overhead of extra header and footer bits as in asynchronous transmission. Finally, Synchronous transmissions are synchronized by an **external clock**, while asynchronous transmissions are synchronized by **special signals along the transmission medium**.

	<b>Advantages</b>	<b>Disadvantages</b>
Asynchronous transmission	<ul style="list-style-type: none"> <li>• Simple, doesn't require synchronization of both communication sides</li> <li>• Cheap, because asynchronous transmission requires less hardware</li> <li>• Setup is faster than other transmissions, so well suited for applications where messages are generated at irregular intervals, for example data entry from the keyboard, and the speed depends on different applications.</li> </ul>	<ul style="list-style-type: none"> <li>• Large relative overhead, a high proportion of the transmitted bits are uniquely for control purposes and thus carry no useful information</li> </ul>
Synchronous transmission	<ul style="list-style-type: none"> <li>• Lower overhead and thus, greater throughput</li> </ul>	<ul style="list-style-type: none"> <li>• Slightly more complex</li> <li>• Hardware is more expensive</li> </ul>