Digital Transmission

Introduction

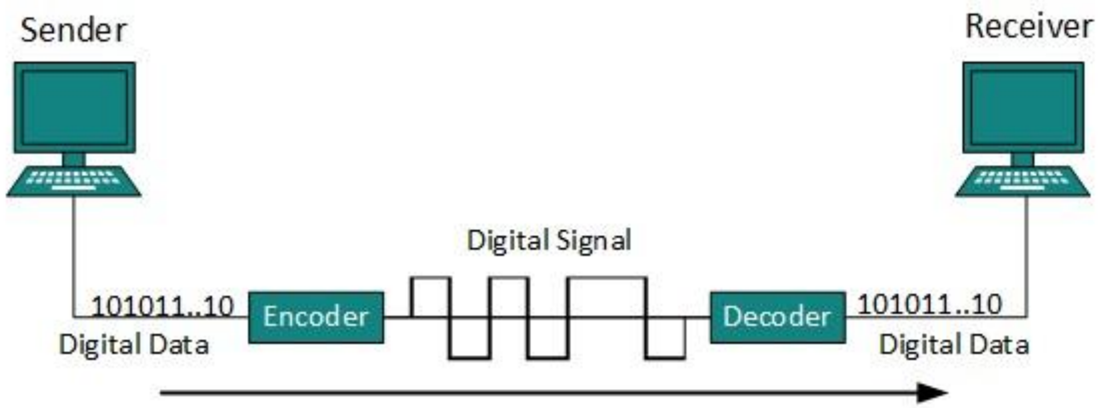
Data or information can be stored in two ways, analog and digital. For a computer to use that data is must be in discrete digital form. Like data, signals can also be in analog and digital form. To transmit data digitally it needs to be first converted to digital form.

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.

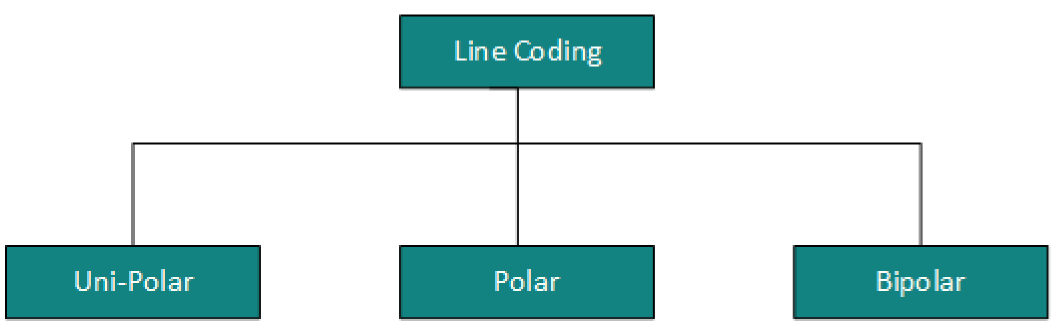
Line Coding

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



[*Image: Line Coding*]

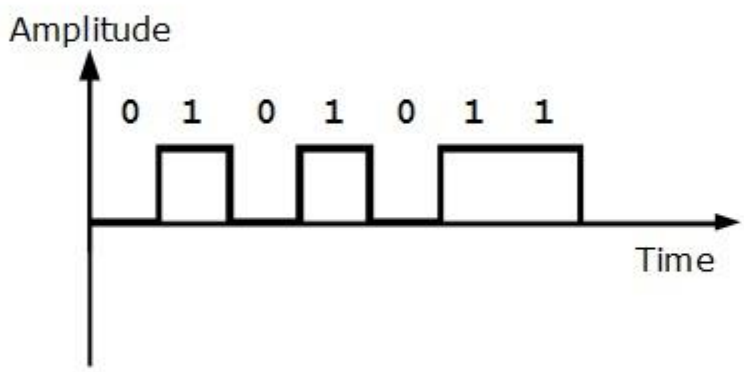
Digital signals which represents digital data, represented as discrete signals. There are three types of line coding schemes available:



[*Image: Line Coding Schemes*]

UNI-POLAR ENCODING

Unipolar encoding schemes uses 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's no rest condition i.e. it either represents 1 or 0.



[*Image: UniPolar NRZ Encoding*]

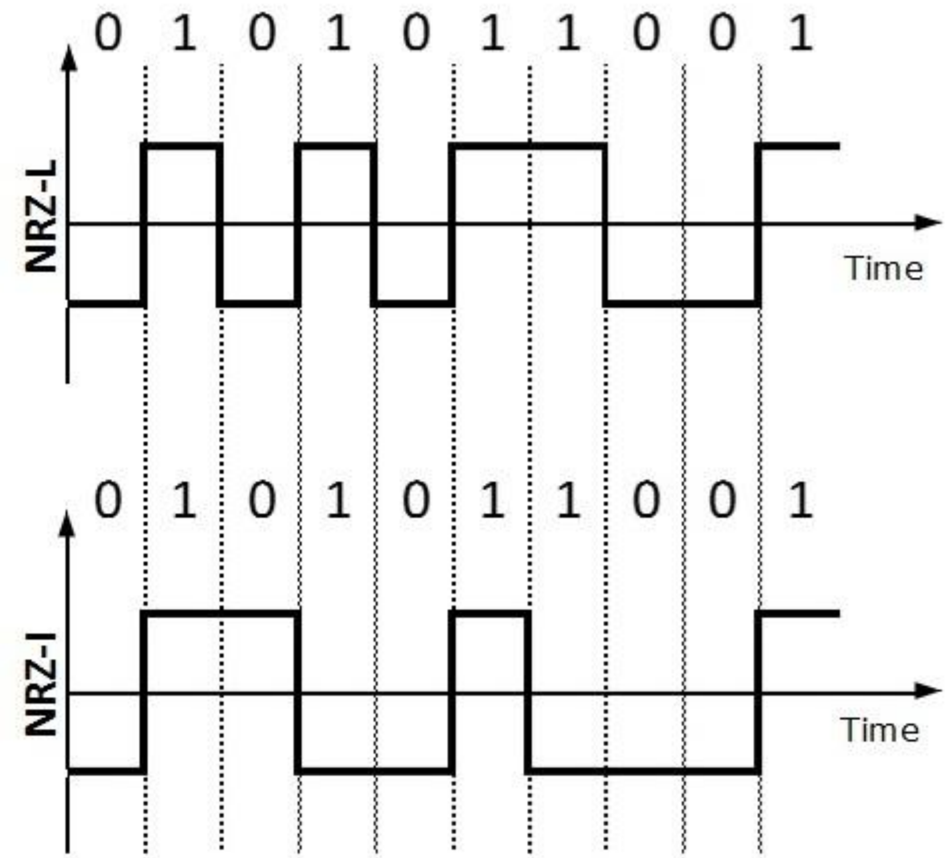
POLAR ENCODING

Polar encoding schemes multiple voltage levels are used to represent binary values. Polar encodings are available in four types:

- POLAR-NRZ (NON-RETURN TO ZERO)

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's no rest condition.

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

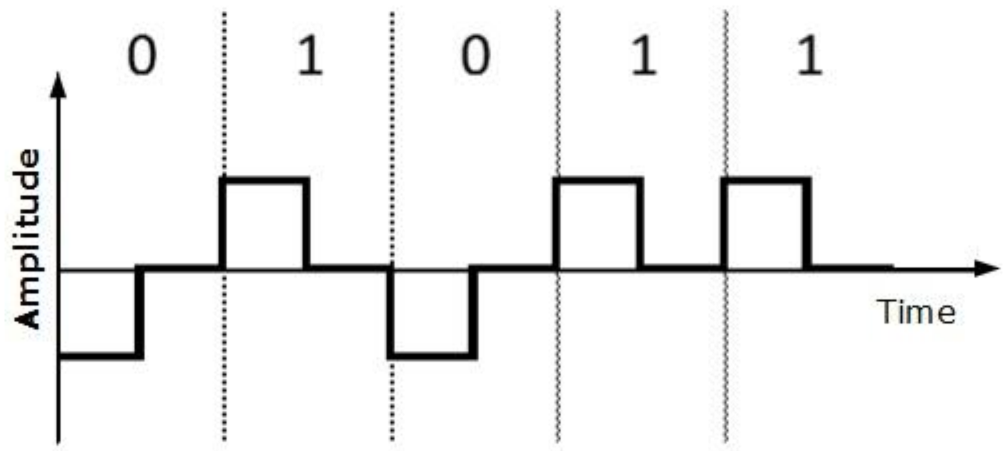


[*Image: NRZ-L and NRZ-I*]

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

- RZ (RETURN TO ZERO)

Problem with NRZ was 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.



[*Image: Return-to-Zero Encoding*]

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

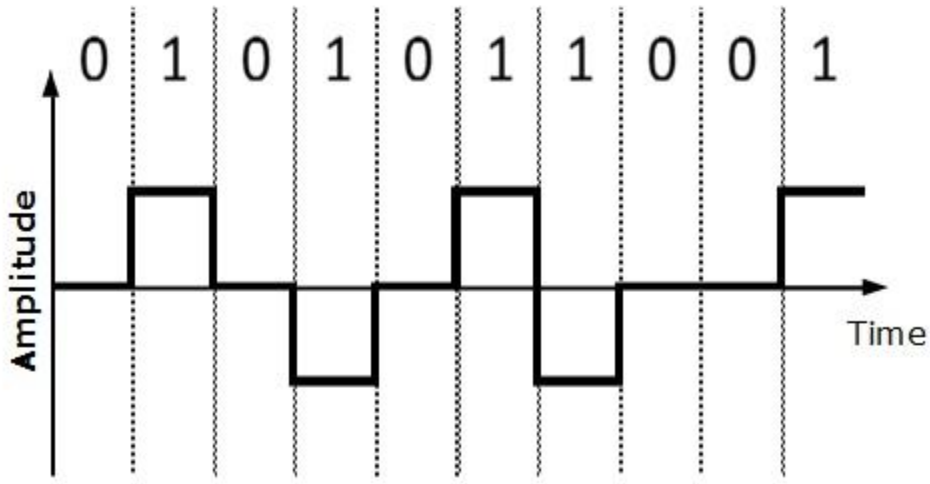
This encoding scheme is a combination of RZ and NRZ-L. Bit time is divided into two halves. It transitions at the middle of the bit and changes phase when a different bit is encountered.

- DIFFERENTIAL MANCHESTER

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

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.



[*Image: Bipolar Encoding*]

Block Coding

To ensure accuracy of data frame received, 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 are increased. It is called Block Coding.

Block coding is represented by slash notation, mB/nB, that is m-bit block is substituted with n-bit block where n > m. Block coding involves three steps: division, substitution and combination.

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

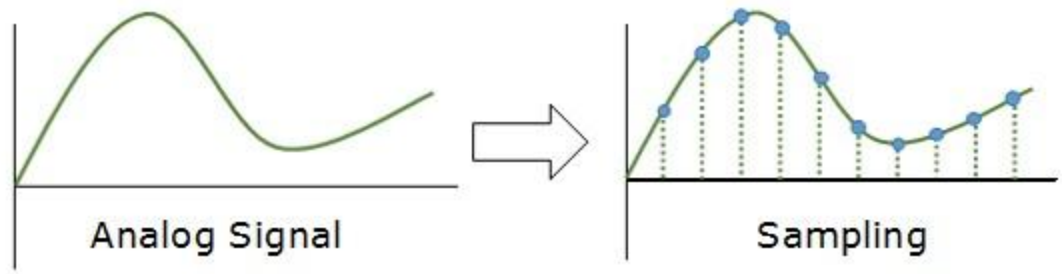
Analog-to-digital conversion

Microphones creates analog voice and camera creates analog videos, which here in our case is treated is analog data. To transmit this analog data over digital signals we need an analog to digital conversion.

Analog data is wave form continuous stream of data whereas digital data is discrete. To convert analog wave into digital data we use Pulse Code Modulation.

Pulse Code Modulation is one of the most commonly used method to convert analog data into digital form. It involves three steps: Sampling, Quantization and Encoding.

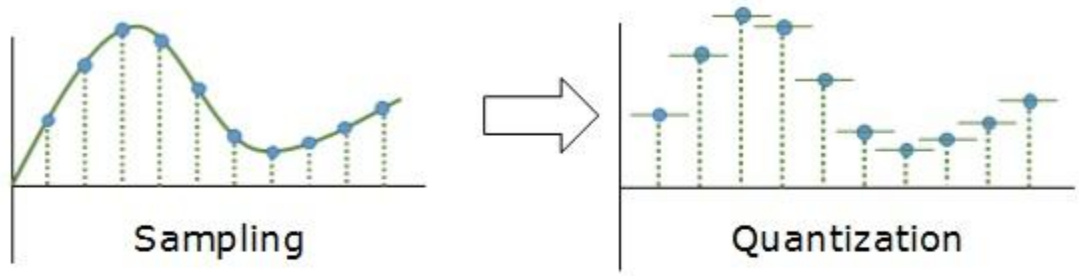
SAMPLING



[*Image: Sampling of Analog Signal*]

The analog signal is sampled every T interval. Most important factor in sampling is the rate on 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.

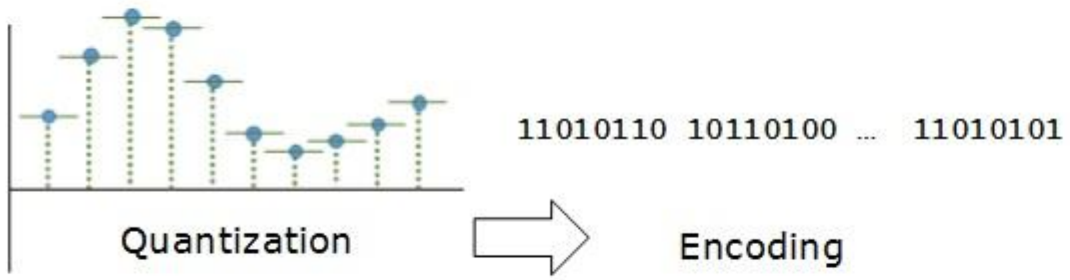
QUANTIZATION



[*Image: Quantization of sampled analog signal*]

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



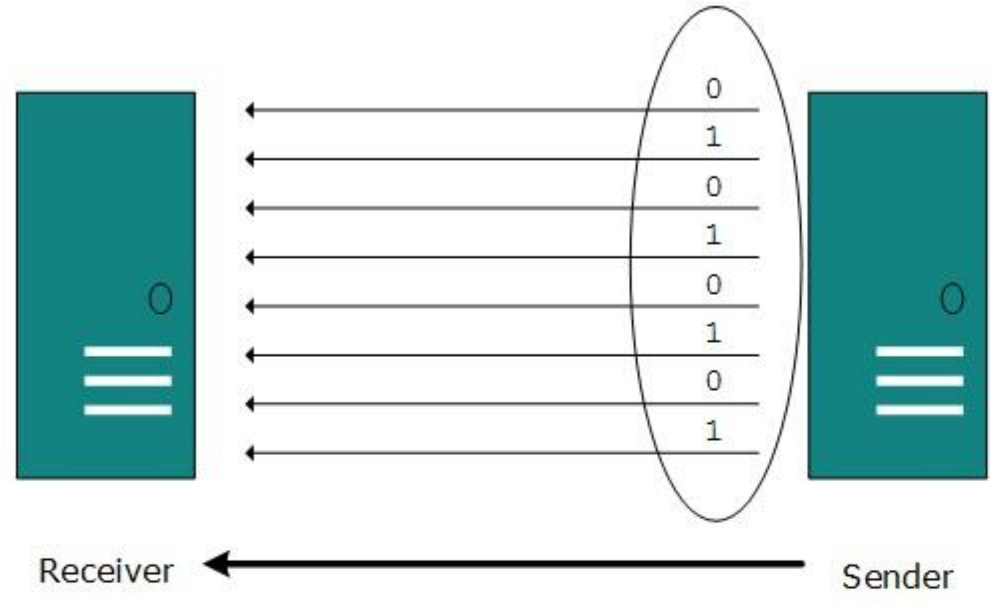
[*Image: Encoding from quantization*]

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

Transmission Modes

How data is to be transferred between to computer is decided by the transmission mode they are using. Binary data i.e. 1s and 0s can be sent in two different modes: Parallel and Serial.

PARALLEL TRANSMISSION

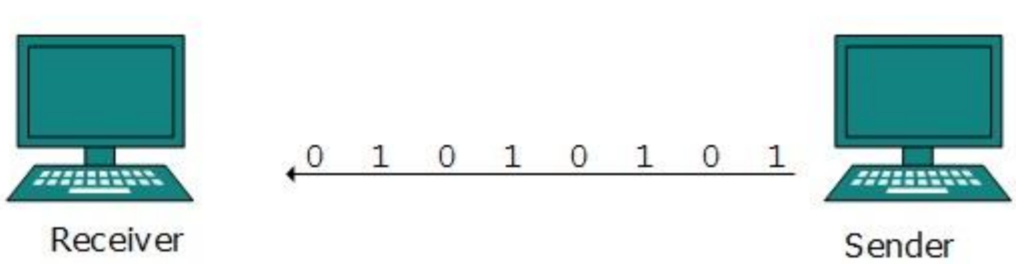


[*Image: Parallel Transmission*]

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 computer distinguish between high order and low order data lines. The sender sends all the bits at once on all lines. Because 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 speed and disadvantage is the cost of wires, as it is equal to the number of bits needs to send parallelly.

SERIAL TRANSMISSION

In serial transmission, bits are sent one after another in a queue manner. Serial transmission requires only one communication channel as oppose parallel transmission where communication lines depends upon bit word length.



[*Image: Serial Transmission*]

Serial transmission can be either asynchronous or synchronous.

ASYNCHRONOUS SERIAL TRANSMISSION

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

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

SYNCHRONOUS SERIAL TRANSMISSION

<="" p="">

It is up to the receiver to recognize and separate bits into bytes. The advantage of synchronous transmission is speed and it has no overhead of extra header and footer bits as in asynchronous transmission.

Source:

http://www.tutorialspoint.com/data_communication_computer_network/digital_transmission.htm