CHAPTER ONE
Data communication Basics

Table of Contents

CHAPTER ONE
Data communication Basics

1.1 Data communication

Data communication is the process of sharing ideas, information and messages with others in a particular time and place. This sharing can be local or remote. Between individuals, local communication usually occurs face to face, while remote communication takes place over distance. The term telecommunication**,** which includes telephony, telegraphy, and television, means communication at a distance ("tele**"** is Greek for "far").

Communication refers to the exchange of data between two or more parties. Parities may refer to person's organization or devices or machines. Computer communication is a process in which two or more computers or devices transfer or transmit data, instructions, and information. In other words, computer data communications is the movement of computer information from one point to another by means of electrical or optical transmission systems (called networks).

The word *data* refers to information/signal/message presented in whatever form is agreed upon by the parties creating and using the data.

Data communication is a subset of telecommunication involving the transmission of data to and from computers and components of computer systems. More specifically, data Communication is the exchange of data (in the form of 0's and 1's) between two or more devices (computers) via some form of the transmission medium such as wires, coaxial cables, fiber optics, or radiated electromagnetic waves such as broadcast radio, infrared light, microwaves, and satellites.

For data communication to occur, the communicating devices must be a part of a communication system made up of some specific kind of hardware and software. This type of a system is known as a data communication system. The effectiveness of a data communications system depends on four fundamental characteristics: delivery, accuracy, timeliness, and jitter.

i. **Delivery**. The system must deliver data to the correct destination. Data must be received by the intended device or user and only by that device or user.
ii. **Accuracy**. The system must deliver the data accurately. Data that have been altered in transmission and left uncorrected are unusable.

iii. **Timeliness**. The system must deliver data in a timely manner. Data delivered late are useless. In the case of video and audio, timely delivery means delivering data as they are produced, in the same order that they are produced, and without significant delay. This kind of delivery is called *real-time* transmission.

iv. **Jitter**. Jitter refers to the variation in the packet arrival time. It is the uneven delay in the delivery of audio or video packets. Abrupt and unwanted variations of one or more signal characteristics, such as the interval between successive pulses

Components of Data Communication System

A data communications system has five components as it is shown on Fig 1.1.(Fig 1.2)

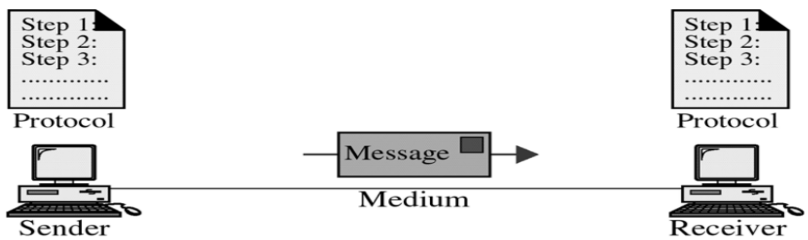


Figure 1 Five components of data communication

A. **Message**. The message is the information (data) to be communicated. Popular forms of information include text, numbers, pictures, audio, and video.

B. **Sender**. The sender is the device that sends the data message. It can be a computer, workstation, telephone handset, video camera, and so on.

C. **Receiver**. The receiver is the device that receives the message. It can be a computer, workstation, telephone handset, television, and so on.

D. **Transmission medium**. The transmission medium is the physical path by which a message travels from sender to receiver. Some examples of transmission media include twisted-pair wire, coaxial cable, fiber-optic cable, and radio waves.

E. **Protocol**. A protocol is a set of rules that govern data communications. It represents an agreement between the communicating devices. Without a protocol, two devices may be connected but not communicating.

A modified communication system

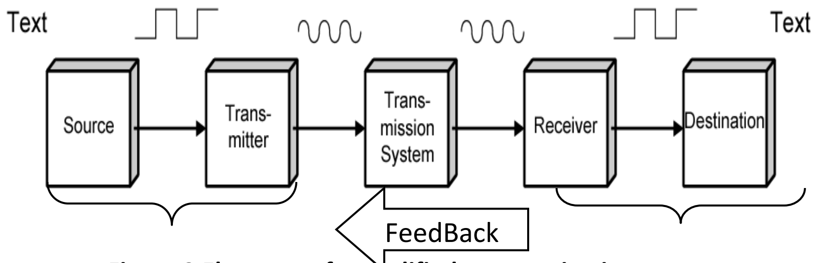


Figure 2 Elements of a modified communication system

1.2 Data representation

Information today comes in different forms such as text, numbers, images, audio, and video.

- ***Text:*** In data communications, text is represented as a bit pattern, a sequence of bits (0s or 1s). Different sets of bit patterns have been designed to represent text symbols. Each set is called a code, and the process of representing symbols is called coding. Today, the prevalent coding system is called Unicode, which uses 32 bits to represent a symbol or character used in any language in the world. The American Standard Code for Information Interchange (ASCII), developed some decades ago in the United States, now constitutes the first 127 characters in Unicode and is also referred to as Basic Latin.
- ***Numbers***: Numbers are also represented by bit patterns. However, a code such as ASCII is not used to represent numbers; the number is directly converted to a binary number to simplify mathematical operations.
- ***Images:*** Images are also represented by bit patterns. In its simplest form, an image is composed of a matrix of pixels (picture elements), where each pixel is a small dot. The size of the pixel depends on the *resolution.* After an image is divided into pixels, each pixel is assigned a bit pattern. The size and the value of the pattern depend on the image. For an image made of only black-and-white dots (e.g., a chessboard), a 1-bit pattern is enough to represent a pixel.
- ***Audio***: Audio refers to the recording or broadcasting of sound or music. Audio is by nature different from text, numbers, or images. It is continuous, not discrete. Even when we use a microphone to change voice or music to an electric signal, we create a continuous signal.
- ***Video***: Video refers to the recording or broadcasting of a picture or movie. Video can either be produced as a continuous entity (e.g., by a TV camera), or it can be a combination of images, each a discrete entity, arranged to convey the idea of motion

Data flow

Communication between two devices can be simplex, half-duplex, or full-duplex.

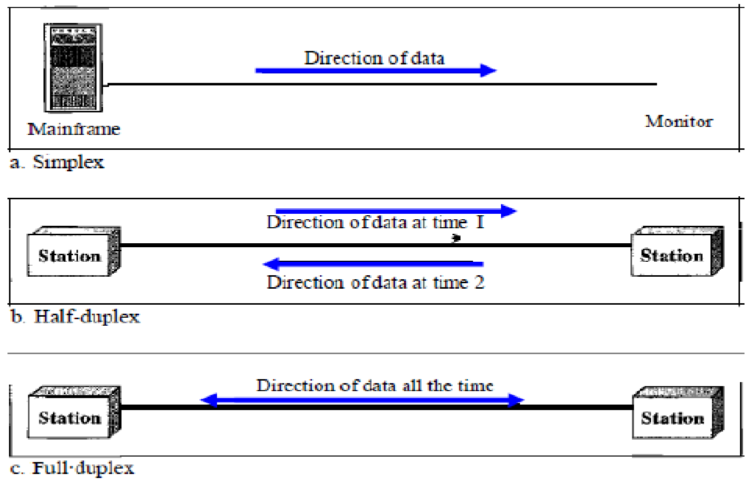


Figure 3 Data flow modes

- In **simplex mode,** the communication is unidirectional. Only one of the two devices on a link can transmit; the other can only receive. The simplex mode can use the entire capacity of the channel to send data in one direction. An example of simplex transmission is the signal sent from the cable TV station to the home television or Radio transmission. If the receiver wants to send feed back to the sender it uses another communication medium, Such as phone or other means of communication.
- In **half-duplex mode,** each station can both transmit and receive, but not at the same time. When one device is sending, the other can only receive, and vice versa. In a half-duplex transmission, the entire capacity of a channel is taken over by whichever of the two devices is transmitting at the time. The half-duplex mode is used in cases where there is no need for communication in both directions at the same time; the entire capacity of the channel can be utilized for each direction. Walkie-talkies and CB (citizens band) radios are both half-duplex systems.
- In **full-duplex mode** (also called *duplex*), both stations can transmit and receive simultaneously. In full-duplex mode, signals going in one direction share the capacity of the link with signals going in the other direction. This sharing can occur in two ways: Either the link must contain two physically separate transmission paths, one for sending and the other for receiving; or the capacity of the channel is divided between signals traveling in both directions. One common example of full-duplex communication is the telephone network. When two people are communicating by a telephone line, both can talk and listen at the same time. The full-duplex mode is used when communication in both directions is required all the time. The capacity of the channel, however, must be divided between the two directions.

1.3 Digital data transmission

In communication system, data is transmitted from one point to another by means or form of electrical signal across a transmission medium. Whether you are collecting numerical statistics from another computer, sending animated pictures from a design workstation, or causing a bell to ring at a distant control center, you are working with the transmission of **data** across network connections.

Generally, the data usable to a person or application are not in a form that can be transmitted over a network. For example, a photograph must first be changed to a form that transmission media can accept. Transmission media work by conducting energy along a physical path. For transmission, data needs to be changed to signals.

There are two types of signals that can travel over varies transmission media. These signals are: Analog signals and Digital signals.

- **Analogue signal**: - is continuously varying electromagnetic waves that may be propagated over a variety of media and what we encounter every day of our life. Speech is analogue signal, and varies in amplitude (volume) and frequency (pitch). Main characteristics of analogue signals are amplitude, frequency and phase.
- **Digital signal**: - are the languages of modern-day computers. Digital signals normally comprise only two states. These are expressed as on or off, 1 or 0 respectively.

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. To transmit data digitally or analogically it needs to be first converted to digital or analog form.

Periodic and Nonperiodic Signals

Both analog and digital signals can take one of two forms: *periodic* or *nonperiodic* (sometimes refer to as *aperiodic*, because the prefix *a* in Greek means "non").

A periodic signal completes a pattern within a measurable time frame, called a period, and repeats that pattern over subsequent identical periods. The completion of one full pattern is called a cycle. A nonperiodic signal changes without exhibiting a pattern or cycle that repeats over time.

Both analog and digital signals can be periodic or nonperiodic. In data communications, we commonly use periodic analog signals (because they need less bandwidth) and nonperiodic digital signals (because they can represent variation in data)

1.4 Signal Encoding Techniques

Basic terminologies

Data rate/bit rate - defines the number of data elements (bits) sent in one second (1s). The unit is bits per second (bps). Most digital signals are nonperiodic/aperiodic, and thus period and frequency are not appropriate characteristics. Two new terms, bit interval (instead of period) and bit rate (instead of frequency) are used to describe digital signals. The bit interval is the time required to send one single bit. The bit rate is the number of bit interval per second. This mean that the bit rate is the number of bits send in one second, usually expressed in bits per second

Signal is electrical, electronic or optical representation of data, which can be sent over a communication medium. Stated in mathematical terms, a signal is merely a function of the data. For example, a microphone converts voice data into voice signal, which can be sent over a pair of wire. **Signal rate-** the number of signal elements sent in 1s. The unit is the baud. The signal rate is sometimes called the **pulse rate,** the **modulation rate,** or the **baud rate.**

Amplitude refers to the strength of a signal, usually measured in volts.

Period refers to the amount of time, in seconds, a signal needs to complete 1 cycle.

Frequency refers to the number of periods in 1 s. Note that period and frequency are just one characteristic defined in two ways. Period is the inverse of frequency, and frequency is the inverse of period i.e. ***f=1/T and T=1/f.*** Frequency is formally expressed in Hertz (Hz), which is cycle per second. Frequency is the rate of change with respect to time. Change in a short span of time means high frequency. Change over a long span of time means low frequency.

Phase/phase shift - describes the position of the signal relative to time 0. If we think of the signal as something that can be shifted backward or forward along the time axis, phase describes the amount of that shift. It indicates the status of the first cycle.

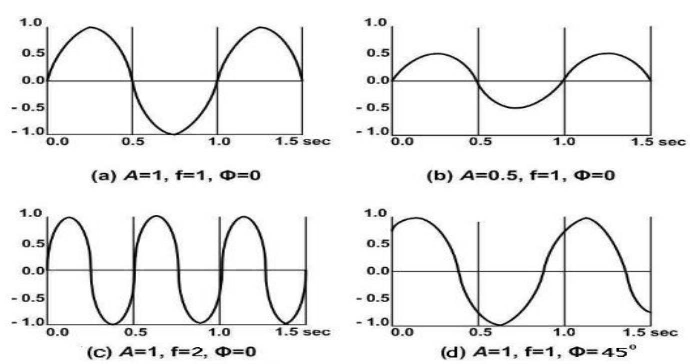


Figure 4 Examples of signals with different amplitude(A), frequency) f) and phase(Ø)

Bandwidth- The range of frequencies contained in a composite signal is its bandwidth**.** The bandwidth is normally a difference between two numbers. For example, if a composite signal contains frequencies between 1000 and 5000, its bandwidth is 5000 − 1000, or 4000Hz.

One goal in data communications is to increase the data rate while decreasing the signal rate. Increasing the data rate increases the speed of transmission; decreasing the signal rate decreases the bandwidth requirement.

In networking, we use the term bandwidth in two contexts.

- The first, bandwidth in hertz, refers to the range of frequencies in a composite signal or the range of frequencies that a channel can pass.
- The second, bandwidth in bits per second, refers to the speed of bit transmission in a channel or link.

Bandpass channel — a channel with a bandwidth that does not start from zero. This type of channel is more available than a low-pass channel. If the available channel is a bandpass channel, we cannot send the digital signal directly to the channel; we need to convert the digital signal to an analog signal before transmission.

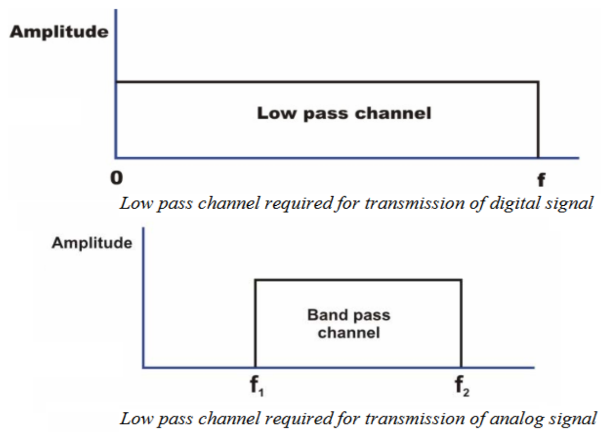


Figure 5 Lowpass and Bandpass

Low-pass channel – is a channel that passes frequencies between 0 and *f*. *Low-pass* refers to a *channel* with a bandwidth that starts from zero. *Low-pass* is used in base-band and *bandpass* is used in broadband. *Low-pass* has one threshold frequency. *Band-pass* has two threshold frequencies. A digital signal can be considered as a signal with an infinite number of frequencies and transmission of digital requires a low-pass channel. On the other hand, transmission of analog signal requires band-pass.

Baseband transmission- means sending a digital signal over a channel without changing the digital signal to an analog signal. using a low-pass channel. For example, the entire bandwidth of a cable connecting two computers is one single channel

Broadband transmission- is a transmission of signals using modulation of a higher frequency signal. The term implies a wide-bandwidth data combined from different sources. On the other hand, broadband systems are those, which use analog signaling to transmit information using a carrier of high frequency.

Digital data to digital signal conversion

In this part we are going to see how to convert digital data into digital signals. It can be done in three ways; line coding, block coding and scrambling. For all communications, line coding is necessary whereas block coding and scrambling are not mandatory.

A. 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 a binary bit. It is represented (stored) internally as series of 1s and 0s. At the sender, digital data are encoded into a digital signal; at the receiver, the digital data are recreated by decoding the digital signal.

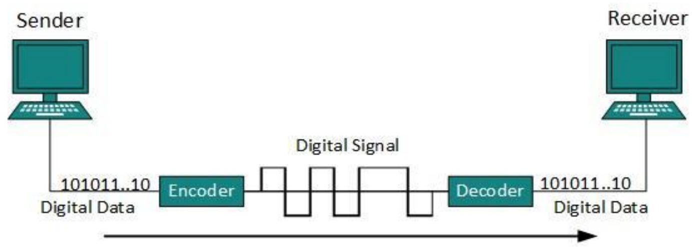


Figure 6 Line coding and decoding

Digital signals which represents digital data, represented as discrete signals. Three types of line coding schemes are available: unipolar, polar and bi-polar.

i. Unipolar encoding

Unipolar encoding scheme 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 is no rest condition i.e. it either represents 1 or 0.4

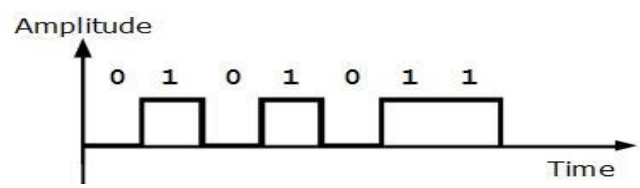


Figure 7 Unipolar NRZ Encoding

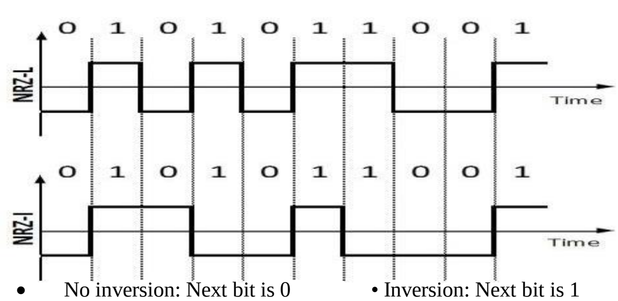


Figure 8 Polar-NRZ schemes: NRZ-L and NRZ-I

ii. Polar encoding

In polar encoding schemes multiple voltage levels are used to represent binary values. In polar schemes, the voltages are on the both sides of the time axis. For example, the voltage level for 0 can be positive and the voltage level for I can be negative. Polar encodings are available in four types:

a. 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 is no rest condition. NRZ scheme has two variants: NRZ-L ((NRZ-Level)) where the level of the voltage determines the value of the bit; and NRZ-I (NRZ-Invert) where the change or lack of change in the level of the voltage c

Figure 9 Polar-NRZ (Non-Return to Zero)

change, the bit is 0; if there is a change, the bit is 1.

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

b. 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 receive is clock are not synchronized.

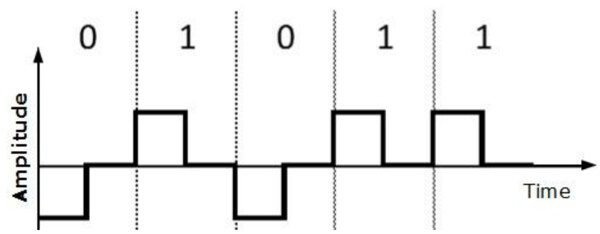


Figure 10 Return-to-Zero Encoding

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.

c. Manchester

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

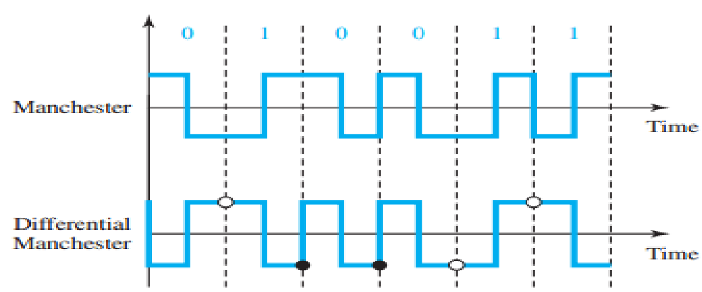


Figure 11 Manchester and Differential Manchester encoding

d. Differential Manchester

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

iii. Bipolar encoding

Bipolar encoding uses three voltage levels, positive, negative and zero. There are two schemes available for bipolar encoding: AMI and Pseudoternary.

a. **AMI** (Alternate Mark Inversion)

Zero voltage represents binary 0 and bit 1 is represented by altering positive and negative voltages.

b. Pseudoternary

Zero voltage represents binary 1 and bit 0 is represented by altering positive and negative voltages.

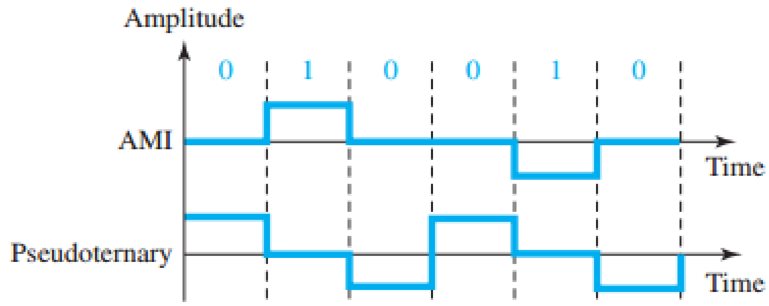


Figure 12 Bipolar schemes: AMI and Pseudoternary

B. 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 numbers 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 (4b/5b).

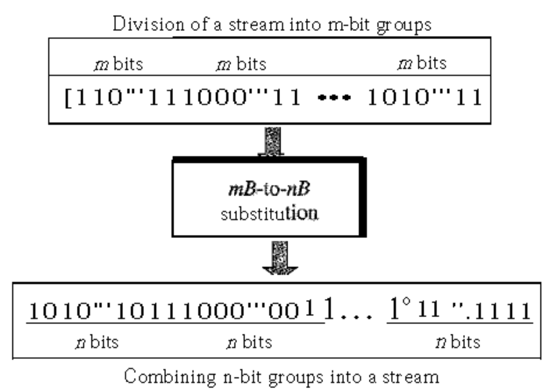


Figure 13 Block Coding

Block coding normally involves three steps: division, substitution, and combination. In the division step, a sequence of bits is divided into groups of *m* bits. For example, in 4B/5B encoding, the original bit sequence is divided into 4-bit groups. The heart of block coding is the substitution step. In this step, we substitute an m-bit group for an n-bit group. For example, in 4B/5B encoding we substitute a 4-bit code for a 5-bit group. Finally, the n-bit groups are combined together to form a stream.

C. Scrambling

Biphase schemes that are suitable for dedicated links between stations in a LAN are not suitable for long-distance communication because of their wide bandwidth requirement. The combination of block coding and

NRZ line coding is not suitable for long-distance encoding either, because of the DC component. Bipolar AMI encoding, on the other hand, has a narrow bandwidth and does not create a DC component. However, a long sequence of 0s upsets the synchronization. If we can find a way to avoid a long sequence of 0s in the original stream, we can use bipolar AMI for long distances. We are looking for a technique that does not increase the number of bits and does provide synchronization. We are looking for a solution that substitutes long zero-level pulses with a combination of other levels to provide synchronization. One solution is called scrambling.

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

A. Pulse Code Modulation

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

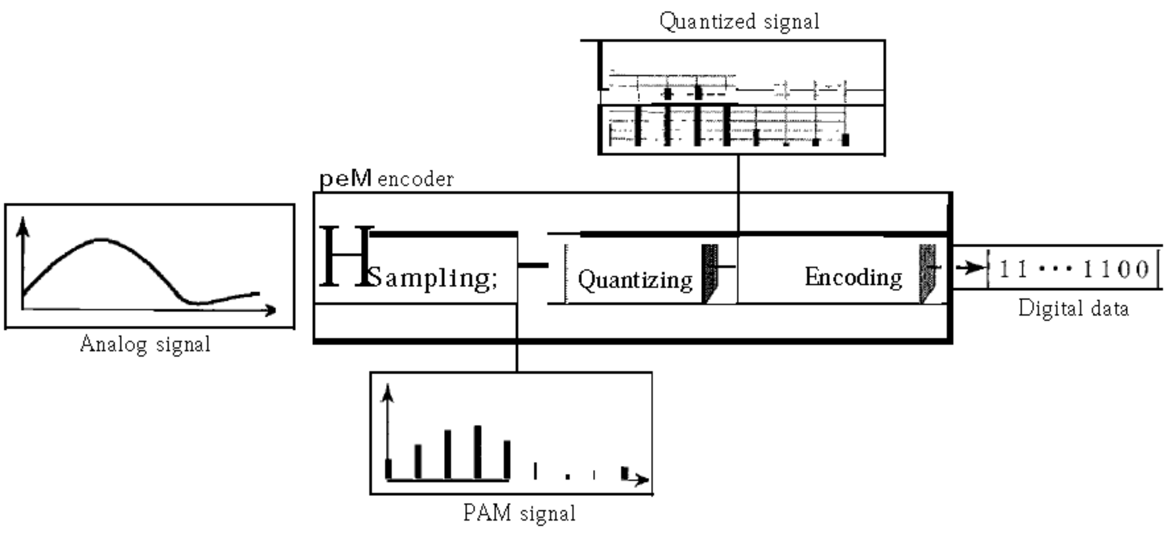


Figure 14 Pulse Code Modulation

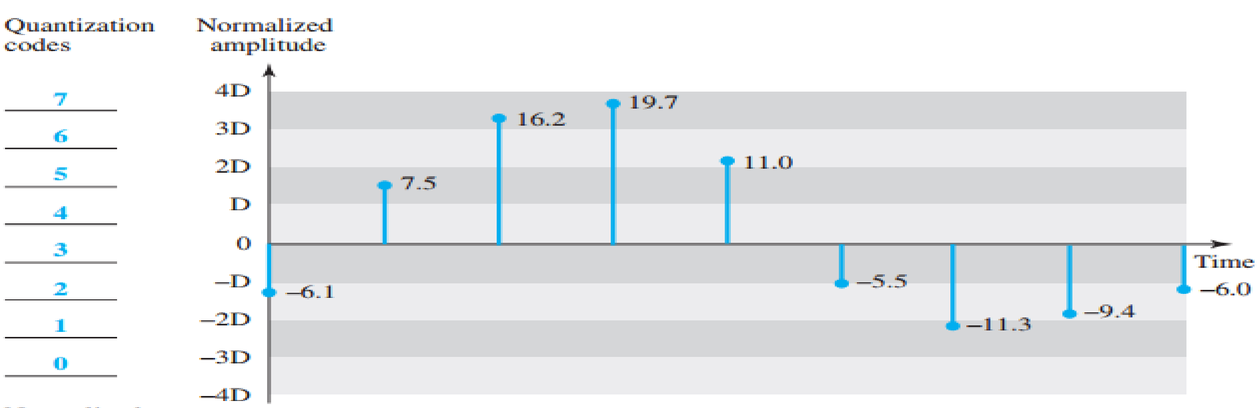
Sampling: The analog signal is sampled every T interval. Most important factor in sampling is the rate on which analog signal is sampled. According to Nyquist Theorem, the sampling rate must be at least two times of the highest frequency of the signal. To elaborate the theorem, first, we can sample a signal only if the signal is band-limited. In other words, a signal with an infinite bandwidth cannot be sampled. Second, the sampling rate must be at least 2 times the highest frequency, not the bandwidth. If the analog signal is low-pass, the bandwidth and the highest frequency are the same value. If the analog signal is bandpass, the bandwidth value is lower than the value of the maximum frequency. Nyquist sampling rate: f_s=2f; *where f is the highest frequency.*

Quantization: is a method of assigning integral values in a specific range to sampled instances. The result of sampling is a series of pulses with amplitude values between the maximum and minimum amplitudes of the signal. The set of amplitudes can be infinite with nonintegral values between the two limits. These values cannot be used in the encoding process. The following are the steps in quantization:

1. We assume that the original analog signal has instantaneous amplitudes between *V*min and *V*max.
2. We divide the range into *L* zones, each of height Δ (delta). Δ=(*V*max - *V*min)/L
3. We assign quantized values of 0 to *L*−1 to the midpoint of each zone.
4. We approximate the value of the sample amplitude to the quantized values.

As a simple example shown in Fig 1.9, assume that we have a sampled signal and the sample amplitudes are between −20 and +20 V. We decide to have eight levels (*L* = 8). This means that Δ = 5V.

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



Normalized PAM values	−1.22	1.50	3.24	3.94	2.20	−1.10	−2.26	−1.88	−1.20
Normalized quantized values	−1.50	1.50	3.50	3.50	2.50	−1.50	−2.50	−1.50	−1.50
Normalized error	−0.38	0	+0.26	−0.44	+0.30	−0.40	−0.24	+0.38	−0.30
Quantization code	2	5	7	7	6	2	1	2	2
Encoded words	010	101	111	111	110	010	001	010	010

Figure 15 Quantization and encoding of a sampled signal

- Normalized PAM values=(Actual amplitude/Δ)

- Normalized quantized values=quantization value from the middle of each zone
- Normalized error= Normalized quantized values- Normalized PAM values

(Read Behrouz A. Forouzan.(2013).Data communication and networkings,5th edition. Section 4.2.1)

B. Delta Modulation

PCM is a very complex technique whereas DM is the simplest modulation technique. PCM finds the value of the signal amplitude for each sample; DM finds the change from the previous sample.

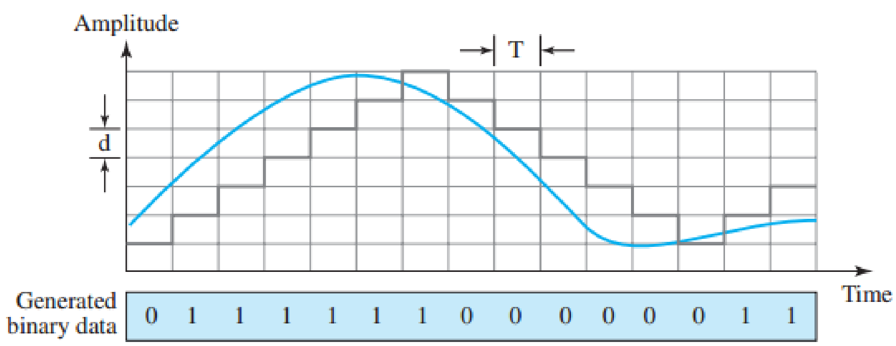


Figure 16 Delta modulation

Digital-to-Analog Conversion

When data from one computer is sent to another via some analog carrier, it is first converted into analog signals. Analog signals are modified to reflect digital data, i.e. binary data. An analog is characterized by its amplitude, frequency and phase. There are three kinds of digital-to-analog conversions possible:

A. Amplitude shift keying

In this conversion technique, the amplitude of analog carrier signal is modified to reflect binary data.

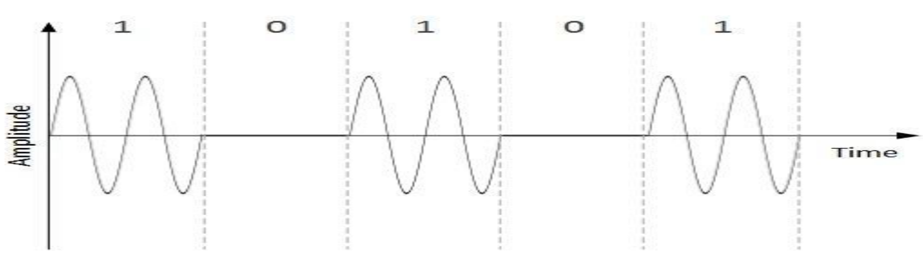


Figure 17 Amplitude Shift Keying

When binary data represents digit 1, the amplitude is held otherwise it is set to 0. Both frequency and phase remain same as in the original carrier signal.

B. Frequency shift keying

In this conversion technique, the frequency of the analog carrier signal is modified to reflect binary data.

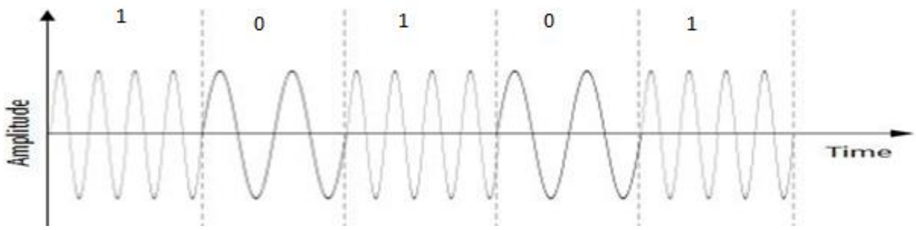


Figure 18 Frequency shift keying

This technique uses two frequencies, f1 and f2. One of them, for example f1, is chosen to represent binary digit 1 and the other one is used to represent binary digit 0. Both amplitude and phase of the carrier wave are kept intact.

C. Phase shift keying

In this conversion scheme, the phase of the original carrier signal is altered to reflect the binary data.

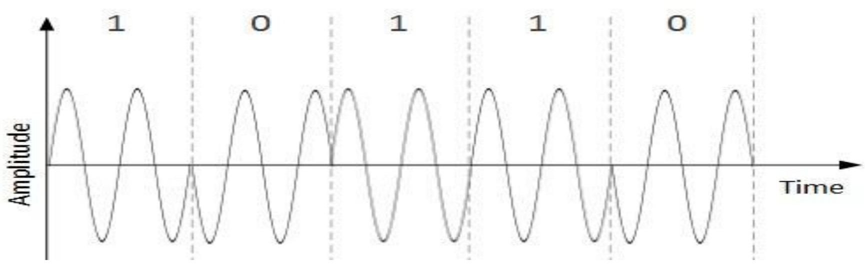


Figure 19 Phase shift keying

When a new binary symbol is encountered, the phase of the signal is altered. Amplitude and frequency of the original carrier signal is kept intact.

D. Quadrature phase shift keying

QPSK alters the phase to reflect 2 binary digits at once. This is done in two different phases. The main stream of binary data is divided equally into two sub-streams. The serial data is converted in to parallel in both sub-streams and then each stream is converted to digital signal using NRZ technique. Later, both the digital signals are merged together.

1.1.1 Analog-to-analog conversion

Analog signals are modified to represent analog data. This conversion is also known as Analog Modulation. Analog modulation is required when bandpass is used. Analog to analog conversion can be done in three ways: Amplitude, frequency and phase modulation.

A. Amplitude modulation

In this modulation, the amplitude of the carrier signal is modified to reflect the analog data.

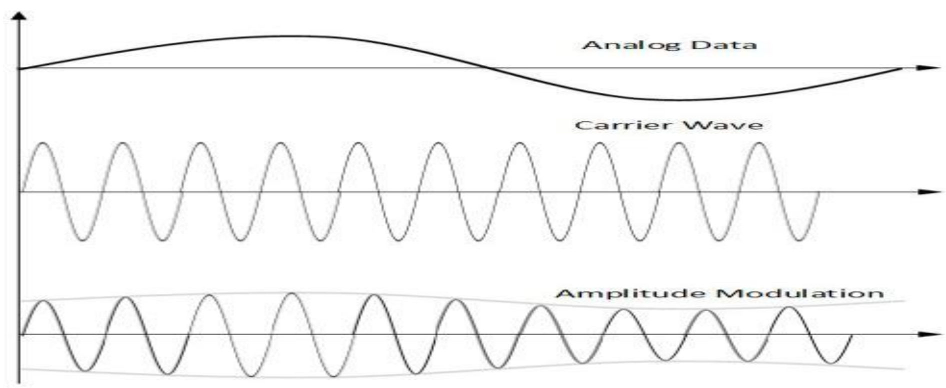


Figure 20 Amplitude Modulation

Amplitude modulation is implemented by means of a multiplier. The amplitude of modulating signal (analog data) is multiplied by the amplitude of carrier frequency, which then reflects analog data. The frequency and phase of carrier signal remain unchanged.

B. Frequency modulation

In this modulation technique, the frequency of the carrier signal is modified to reflect the change in the voltage levels of the modulating signal (analog data).

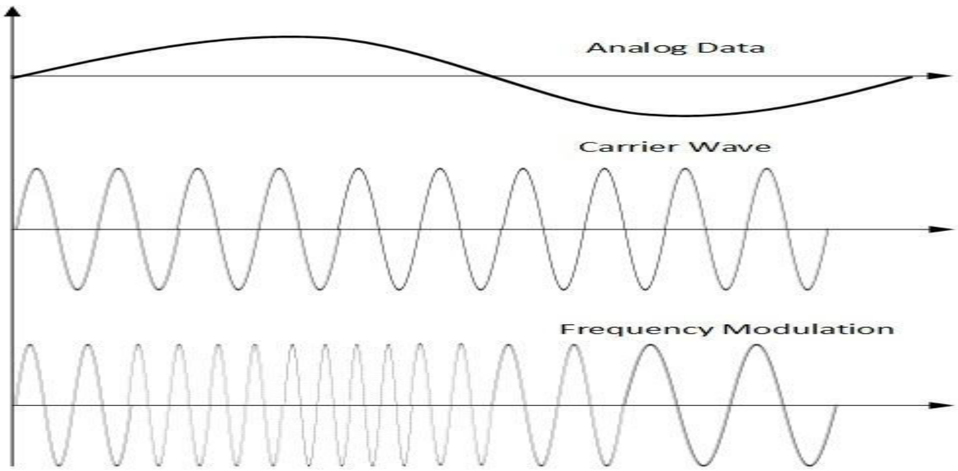


Figure 21 Frequency Modulation

The amplitude and phase of the carrier signal are not altered.

C. Phase modulation

In the modulation technique, the phase of carrier signal is modulated in order to reflect the change in voltage (amplitude) of analog data signal.

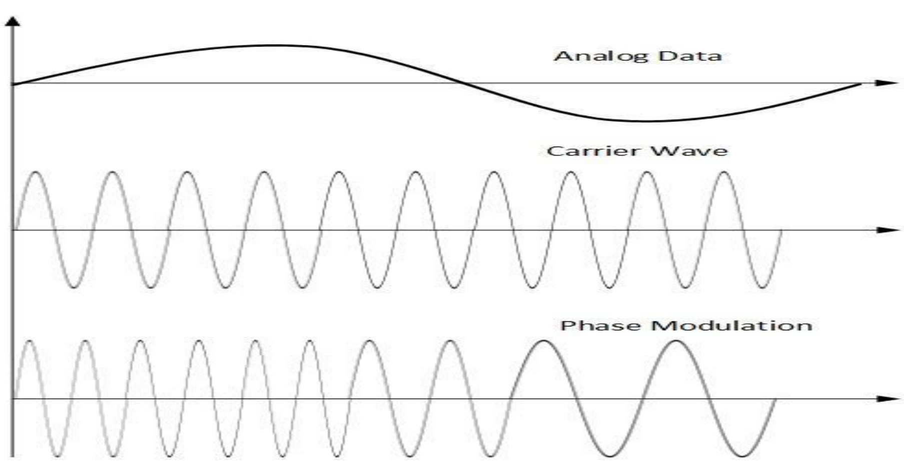


Figure 22 Phase Modulation

Phase modulation practically is similar to Frequency Modulation, but in Phase modulation, frequency of the carrier signal is not increased. Frequency in carrier signal is changed (made dense and sparse) to reflect voltage change in the amplitude of modulating signal.

Modes of Data Transmission

Data transmission means movement of the bits over a transmission medium connecting two devices. How data is to be transferred between to computer is decided by the type of data transmission they are using. Binary data i.e. 1s and 0s can be sent in two different ways: Parallel transmission and Serial transmission.

A. Parallel transmission

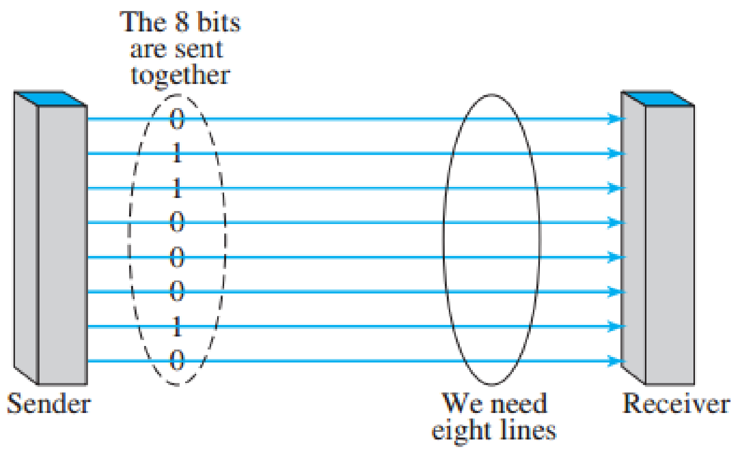


Figure 23 Parallel Transmission

The binary bits are organized into 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 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.

Generally, in this type of transmission all the bits of a *byte* are transmitted simultaneously on separate wires. And it is practicable if two devices are close to each other. E.g. Computer to Printer, Communication within the Computer.

B. Serial transmission

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

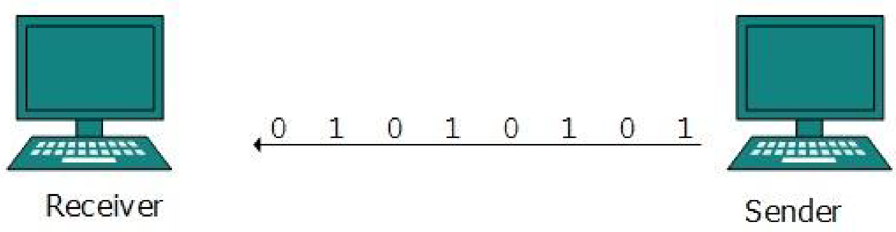


Figure 24 Serial Transmission

In serial transmission bits are transmitted one after the other. Usually the Least Significant Bit (LSB) has been transmitted first. Serial transmission requires only one circuit interconnecting two devices and it is much faster than parallel because of way bits processed (e.g. USB and SATA drives). It is suitable for transmission over Long distance.

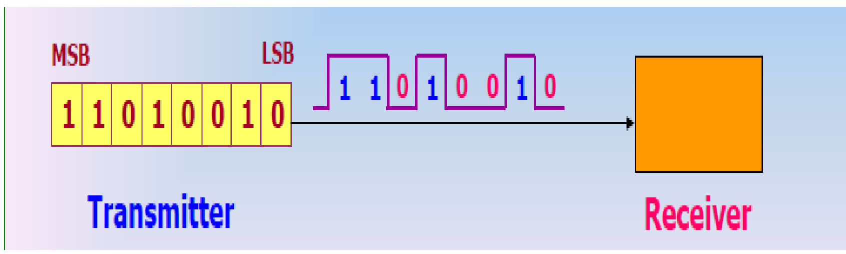


Figure 25 Serial Transmission

Note: Received signal is never same as transmitted. Clock signal samples & regenerates the original bits as it was transmitted. Received signal should be sampled at right instant. Otherwise it will cause bit error.

There are two types of serial transmissions.

A. Asynchronous serial transmission

It is named so because there is 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.

In asynchronous serial transmission

- Sending end commences the transmission of bits at any instant of time.
- Uses stop/ start bits.
- No time relation between the consecutive bits.
- During idle condition Signal ‘1’ is transmitted.

- "Start bit" before the byte and "Stop bit" at the end of the byte for Start/Stop synchronization.
- Most common type of serial data transfer.
- Allows packet switching.
- Allows sharing of bandwidth (i.e. talk on phone while another person is using internet).

B. Synchronous serial transmission

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.

In synchronous serial transmission

- is carried out under the control of the timing source
- no Start/Stop bits
- continuous block of Data are encapsulated with Header & Trailer along with Flags

Why Not use parallel transmission instead of serial transmission?

- Due to inconsistencies on channels data arrives at different times
- Because of the way it is transmitted packet switching cannot be used
- The above two points makes parallel slower than serial and requires higher bandwidth.

Parallel transmissions are rarely used anymore.

Data transmission error detection and correction

Introduction

Networks must be able to transfer data from one device to another with complete accuracy. A system that cannot guarantee that the data received by one device is identical to the data transmitted by another device is essentially useless. Anytime the data is transmitted from source to destination, it gets corrupted on the way. Many factors including noise can alter or wipe out one or more bits of a given data unit. Reliable systems must have a mechanism for detecting and correcting such errors. Error detection and Correction is implemented either at the data link layer or at the transport layer.

1.5 Transmission impairments.

Signals travel through transmission media, which are not perfect. The imperfection causes signal impairment. This means that the signal at the beginning of the medium is not the same as the signal at the end of the medium. What is sent is not what is received.

Main types of transmission impairments are:

- **Attenuation** (decrease, or reduction) Attenuation means a loss of energy. When a signal, simple or composite, travels through a medium, it loses some of its energy in overcoming the resistance of the medium. To compensate for this loss, amplifiers are used to amplify the signal.
- **Distortion** (misrepresentation) means that the signal changes its form or shape. Distortion can occur in a composite signal made of different frequencies. Each signal component has its own propagation speed (see the next section) through a medium and, therefore, its own delay in arriving at the final destination. Differences in delay may create a difference in phase if the delay is not exactly the same as the period duration. In other words, signal components at the receiver have phases different from what they had at the sender
- **Noise –** is another cause of impairment. Several types of noise, such as thermal noise, induced noise, crosstalk, and impulse noise, may corrupt the signal. Thermal noise is the random motion of electrons in a wire which creates an extra signal not originally sent by the transmitter. Induced noise comes from sources such as motors and appliances. These devices act as a sending antenna, and the transmission medium acts as the receiving antenna. Crosstalk is the effect of one wire on the other. One wire acts as a sending antenna \and the other as the receiving antenna. Impulse noise is a spike (a signal with high energy in a very short time) that comes from power lines, lightning, and so on.

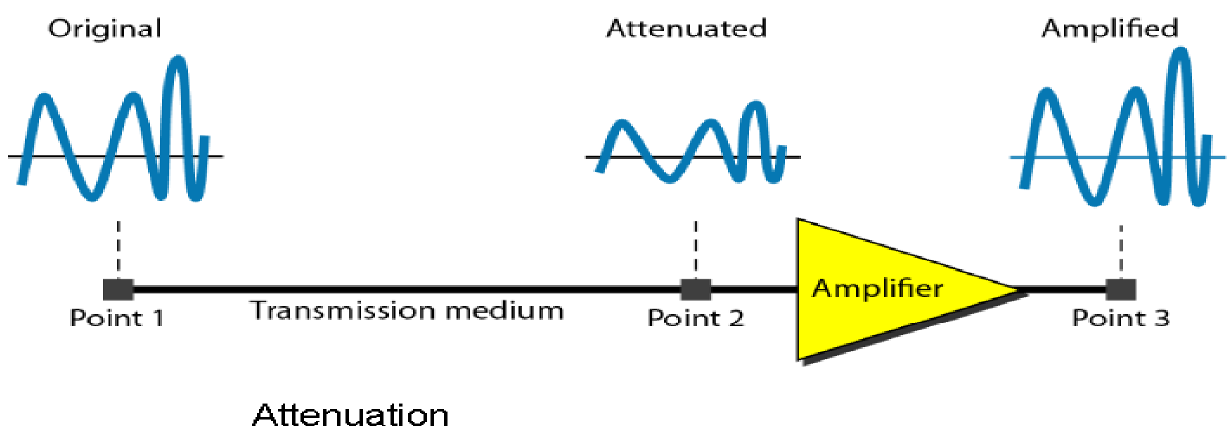


Figure 26 - Attenuation

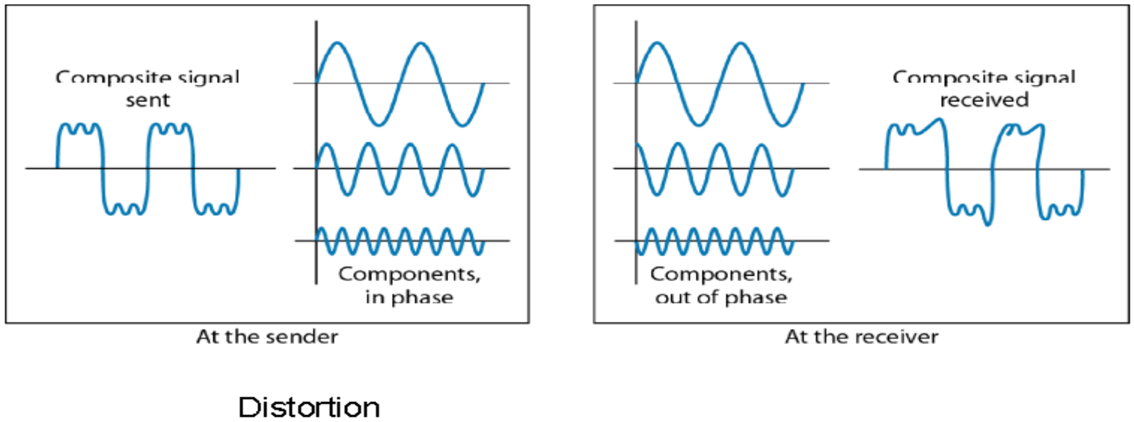


Figure 27 - Distortion

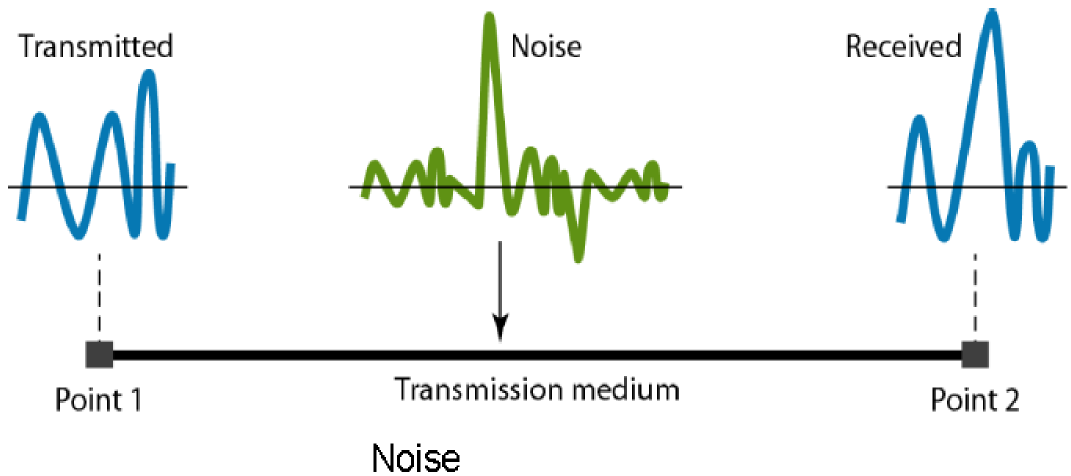


Figure 28 Noise

1.1.2 Type of errors

There are two types of error: single-bit and burst errors. Whenever an electromagnetic signal flows from one point to the other, it is subject to unpredictable interference from heat, magnetism and other forms of electricity. This interference can change the shape or timing of the signal. If the signal is carrying encoded binary data, such changes can alter the meaning of the data.

Single Bit error

In a single-bit error, a 0 is changed to a 1 or a 1 to a 0. For example, a 0.01 second burst of impulse noise on a transmission with a data rate of 1200 bps might change all or some of 12 bits of information.

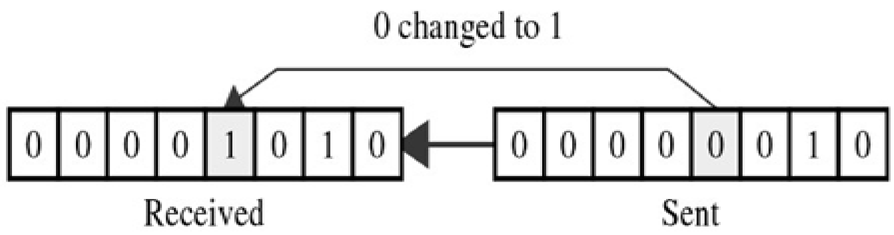


Figure 29 Single bit error

The term single bit error means that only one bit of a given data unit (such as a byte, character, or a packet) is changed from 1 to 0 or from 0 to 1. Figure shows the effect of a single bit error on a data unit. Single bit errors are the least likely type of error in serial data transmission. To see why, imagine a sender sends data at 1Mbps. This means that each bit lasts only 1/1,000,000 seconds or 1 microsecond. For single bit error to occur the noise must have duration of 1 microsecond which is very rare, noise lasts much longer than that. However, single bit error can occur if we are sending data using parallel transmission. For example, if 8 wires are used to send all of the eight bits of a byte at the same time and one of the wires is noise, one bit can be corrupted in each byte.

Burst Errors

The term burst error means that two or more bits in the data unit have changed from 1 to 0 or from 0 to 1.

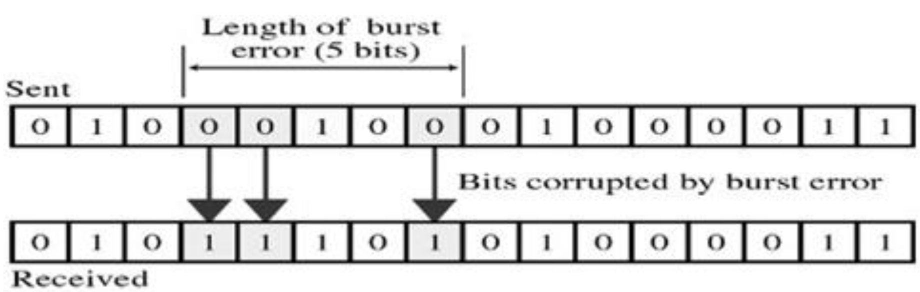


Figure 30 Burst error

The above figure shows the effect of a burst error on a data unit. In this case 0100010001000011 was sent but 0101110101000011 was received. Note that a burst error does not necessarily mean that error occur in consecutive bits.

The length of the burst is measured from the first corrupted bit to the last corrupted bit. Some bits in between may not have been corrupted. Burst error is most likely to happen in a serial transmission. The duration of the noise is normally longer than the duration of a bit which means that when noise affects data, it affects a set of bits. The number of bits affected depends on the data rate and duration of noise.

For example, if we are sending data at 1 Kbps, a noise of 1/100 seconds can affect 10 bits. If we are sending data at 1 Mbps, the same noise can affect 10,000 bits.

1.1.3 Error detection/Redundancy

An error detection technique in which a shorter group of bits may be appended to the end of each data unit is called redundancy. This is because the extra bits are redundant to the information and are discarded as soon as the accuracy of transmission has been determined.

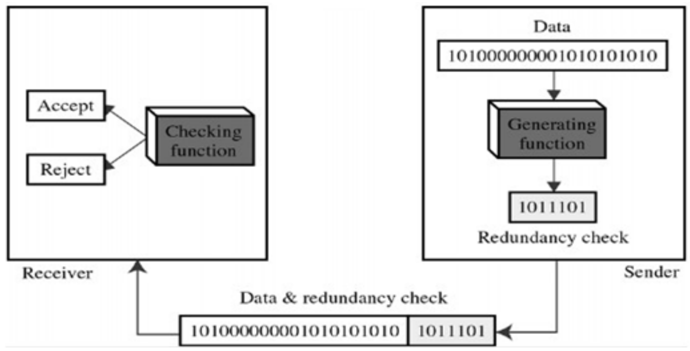


Figure 31 Redundancy

The figure shows the process of using redundant bits to check the accuracy of data unit. Once the data stream has been generated, it passes through a device that analyzes it and adds on an appropriately coded redundancy check. The data unit now enlarged by several bits (7) travels over the link to the receiver. The receiver puts the entire stream through a checking function. If the received bit stream passes the checking criteria, the data portion of the data unit is accepted and the redundant bits are discarded.

Types of Redundancy Checks

There are 4 types of redundancy checks used in data communication:

- Vertical Redundancy Check (VRC)
- Longitudinal Redundancy Check (LRC)
- Cyclic Redundancy Check (CRC)
- Checksum

A. Vertical Redundancy Check (VRC)

- Most common and least expensive
- Also called Parity Check

A redundant bit called parity bit is appended to every data unit so that total number of 1's in the unit becomes even including the parity bit

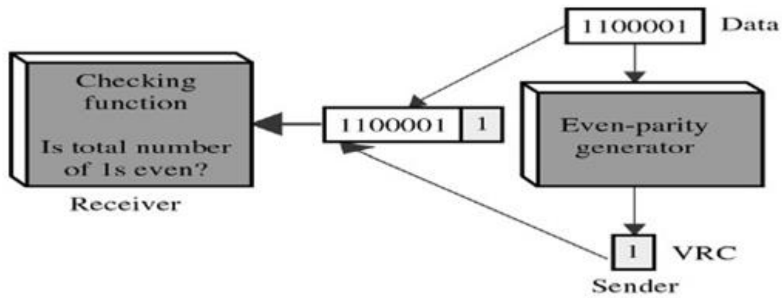


Figure 32 Even parity checking

As it is shown on the figure above, we want to transmit the binary data unit 1100001. Adding together the number of 1's gives us 3, an odd number. Before transmitting, we pass the data unit through a parity generator, which counts the 1's and appends the parity bit (1) to the end. The total number of 1's is now 4, an even number. The system now transfers the entire expanded across the network link. When it reaches its destination, the receiver puts all 8 bits through an even parity checking function. If the receiver sees 11100001, it counts four ones, an even number and the data unit passes. Instead of 11100001, if receiver sees 11100101, then the parity checker counts the 1's, it gets 5 an odd number and therefore rejects the whole unit. Some systems may also use odd parity checking. The principal is the same as even parity.

B. Longitudinal Redundancy Check (LRC)

In LRC, a block of bits is organized in a table (rows and columns). For example, instead of sending 32 bits, we organize them in a table made of 4 rows and 8 columns.

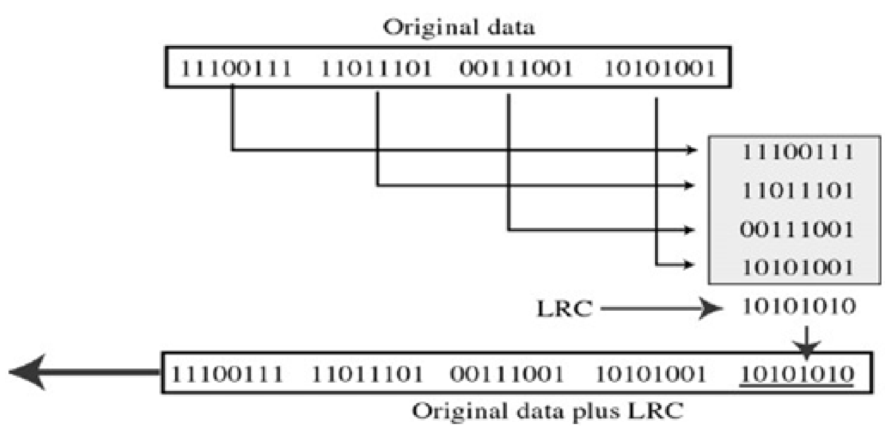


Figure 33 Longitudinal Redundancy Checking

We then calculate the Parity bit for each column and create a new row of 8 bits which are the parity bits for the whole block. Note that the first parity bit in the 5th row is calculated based on all the first bits. The second parity bit is calculated based on all the second bits and so on. We then attach the 8 parity bits to the original data and send them to the receiver.

C. Cyclic Redundancy Check (CRC)

- Most powerful of checking techniques
- VRC and LRC are based on Addition
- CRC is based on Binary Division

A sequence of redundant bits called CRC or CRC remainder is appended to the end of the data unit, so that the resulting data unit becomes exactly divisible by a second predetermined binary number. At its destination, the data unit is divided by the same number. At this step, if there is no remainder, the incoming data unit is assumed to be intact and is therefore accepted. Otherwise, a remainder indicates that a data unit has been damaged and therefore must be rejected.

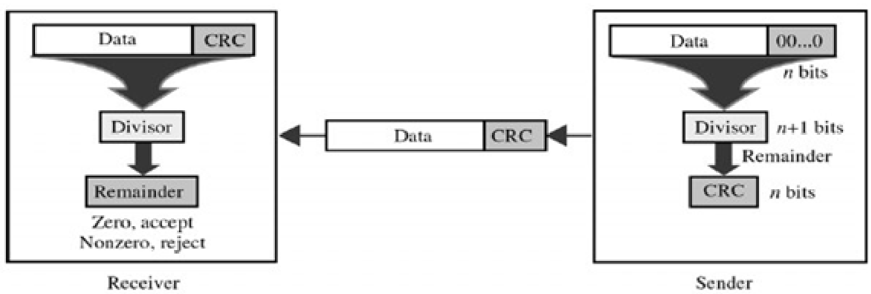


Figure 34 Cyclic Redundancy Checking

First a string of n 0's is appended to the data unit. The number 'n' is one less than the number of bits in the predetermined divisor, which is "n+1" bits. Secondly, newly elongated data unit is divided by the divisor using a process called binary division. The remainder resulting from this division is the CRC. Third, the CRC of 'n' bits replaces the appended 0's at the end of the data unit. Note that CRC may consist of all zeros. The data unit arrives at the receiver followed by the CRC.

The receiver treats the whole string as a unit and divides it by the same divisor that was used to find the CRC remainder. If string arrives without an error, the CRC checker yields a remainder of zero and data unit passes. If the string has been changed in transit, the division yields a non-zero remainder and the data unit does not pass.

D. CHECKSUM

- Error detection method used by the Higher Layers

- Like VRC, LRC, CRC, Checksum is also based on the concept of redundancy

One's Complement

Finding one's complement

- Invert every 1 to 0 and 0 to 1
- A and –A are one's complement of each other
- +A = 1010 → -A = 0101
- +0 = 0000 → -0 = 1111

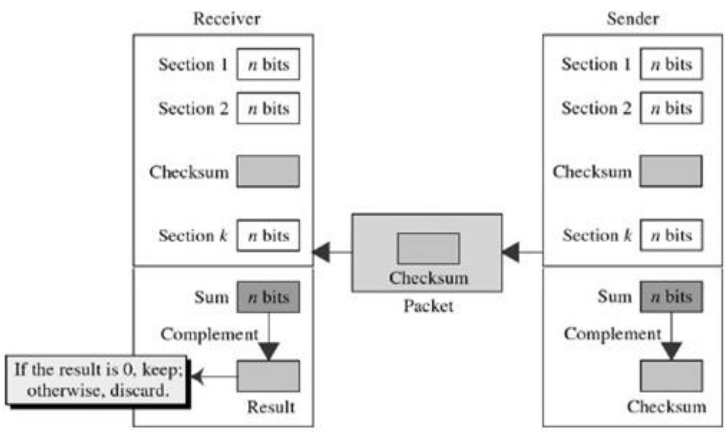


Figure 35 CHECKSUM Generator

The sender subdivides data units into equal segments of 'n' bits (16 bits). These segments are added together using one's complement. The total (sum) is then complemented and appended to the end of the original data unit as redundancy bits called CHECKSUM. The extended data unit is transmitted across the network. The receiver subdivides data unit as above and adds all segments together and complement the result. If the intended data unit is intact, total value found by adding the data segments and the checksum field should be zero. If the result is not zero, the packet contains an error & the receiver rejects it.

1.1.4 Error correction

Error correction locates the invalid bit or bits. Error correction can be done in two ways:

–Receiver can ask sender for retransmission.

–Receiver can use an error-detecting code, which automatically correct certain errors

Error correcting codes are more sophisticated than error detecting codes. They require more redundancy bits and the number of bits required to correct multiple –bit or burst error is so high that in most cases it is inefficient. Error correction is limited to 1, 2 or 3 bit.

Redundancy bit (r)

- r must be able to indicate at least m+r+1 states
- m+r+1 states must be discoverable by r bits
- Therefore, $2^r \geq$ m+r+1
- If m=7, r=4 as $2^4 \geq 7+4+1$

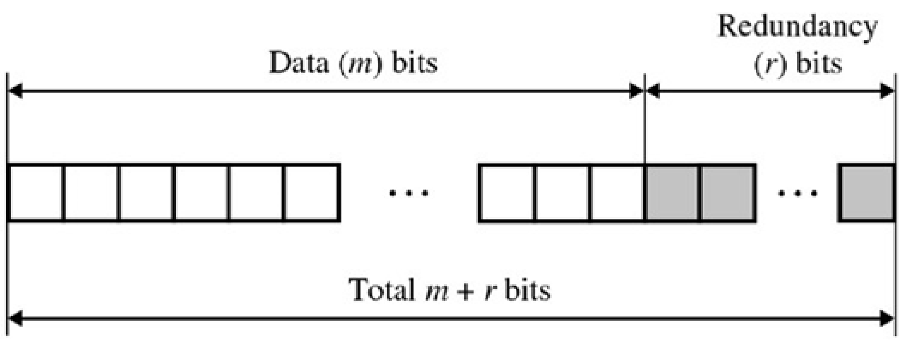
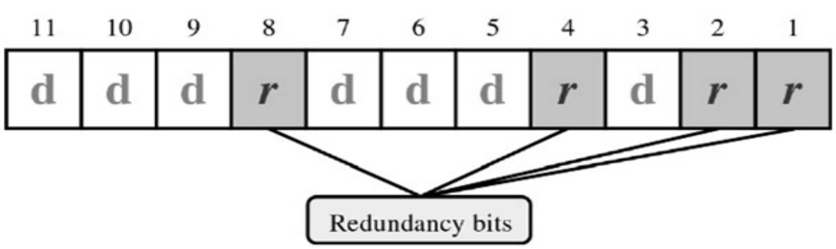
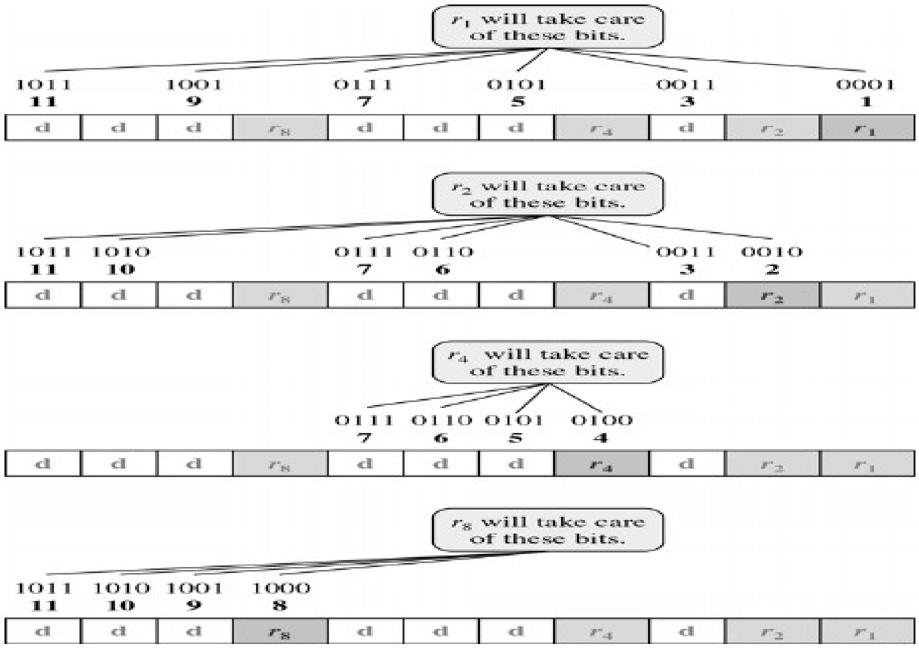


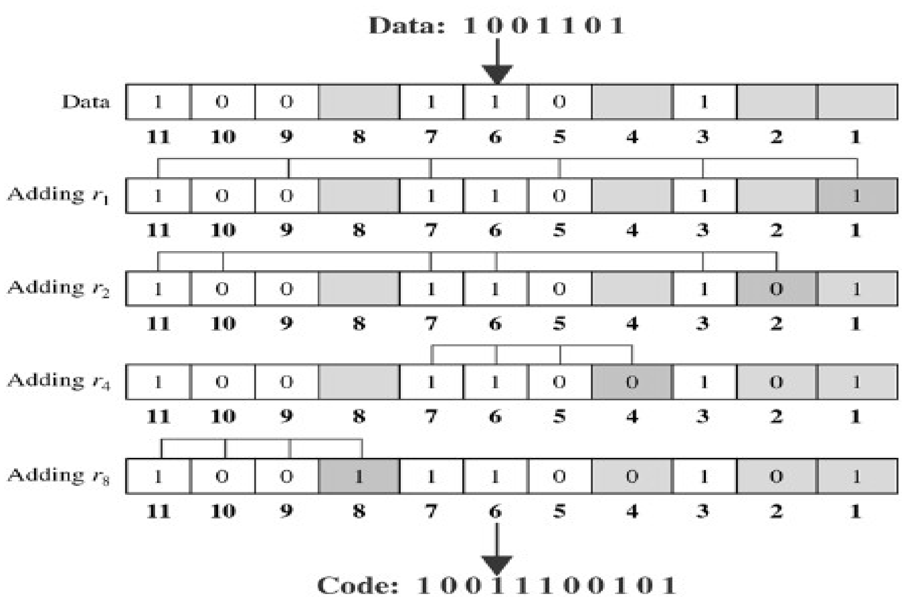
Figure 36 Data plus Redundancy bits

Hamming Code

Each r bit is the VRC bit for one combination of data bits. r1(r2) bit is calculated using all bit positions whose binary representation includes a 1 in the first(second) position, and so on.




r_1 will take care of these bits.
1011 11 1001 9 0111 7 0101 5 0011 3 0001 1
d d d r_8 d d d r_4 d r_2 r_1
r_2 will take care of these bits.
1011 11 1010 10 0111 7 0110 6 0011 3 0010 2
r_4 will take care of these bits.
0111 7 0110 6 0101 5 0100 4
r_8 will take care of these bits.
1011 11 1010 10 1001 9 1000 8


Data: 1 0 0 1 1 0 1
Data
Adding r_1
Adding r_2
Adding r_4
Adding r_8
11 10 9 8 7 6 5 4 3 2 1
Code: 1 0 0 1 1 1 0 0 1 0 1

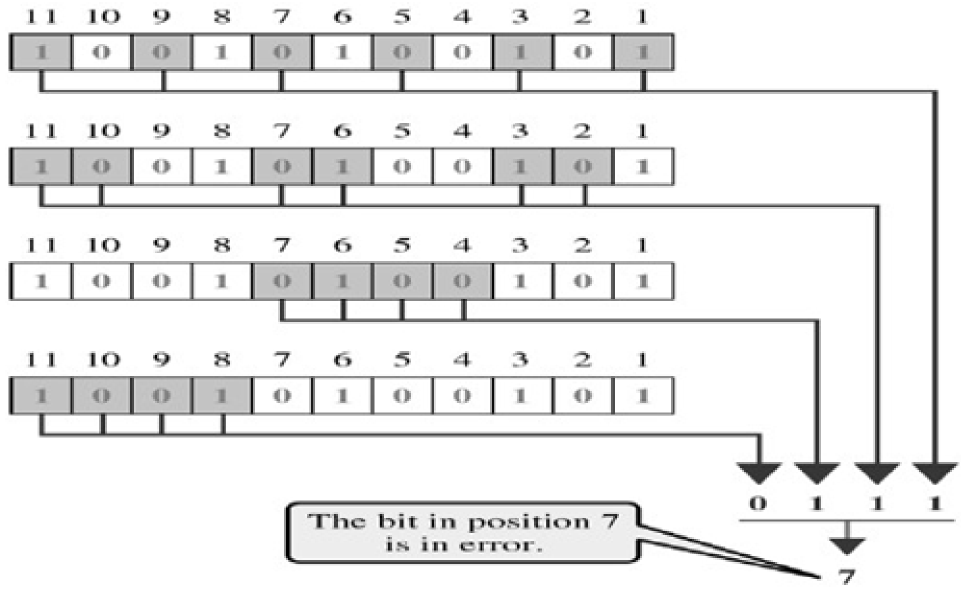


Figure 37 Hamming code recovering a single-bir error

Further reading assignment

1. Discuss the pros and cons of all types of encoding techniques.
2. What is the advantage of Scrambling over line and block coding?
3. Why we need to modulate an analog data to analog signal?
4. Discuss about transmission impairments.
5. Compare error detection techniques in terms of their performance.
6. How does a network detect and correct multiple transmission errors?
7. Is analog transmission or digital transmission preferable on network? Justify.
8. Distinguish between data and signal.
9. What do you mean by a “Periodic Signal”? And what are the three parameters that characterize it?
10. What do you mean by the Bit Interval and Bit rate in a digital signal?